



ŠOLSKI CENTER CELJE

Srednja šola za kemijo, elektrotehniko in računalništvo

Pot na Lavo 22

3000 Celje

SKLADIŠČNI SISTEM

raziskovalna naloga

Mentor:

mag. Boštjan Resinovič, univ. dipl. inž. rač. inf.

Avtorja:

Matjaž Centrih, R-4. b,
Žan Plahuta , R-4. b

Celje, marec 2019

KAZALO VSEBINE

Povzetek	5
Abstract	6
Ključne besede	6
Keywords	6
Kratice in okrajšave.....	7
1 UVOD	8
1.1 Predstavitev problema	8
1.2 Hipoteze.....	8
1.3 Cilji.....	9
1.4 Raziskovalne metode	9
2 SKLADIŠČA IN DELITEV SKLADIŠČ.....	9
2.1 Naloge skladišč.....	10
2.2 Vrste skladišč glede na način gradnje.....	10
2.2.1 Odprto skladišče.....	11
2.2.2 Pokrito skladišče	11
2.2.3 Zaprto skladišče	11
2.2.4 Specialno skladišče	11
2.3 Vrste skladišč glede na vrsto blaga	11
2.4 Glavni skladiščni procesi.....	12
2.4.1 Prevzem blaga	12
2.4.2 Skladiščenje blaga	12
2.4.3 Odprema blaga	12
2.4.4 Komisioniranje blaga	12
3 PROGRAMSKA OPREMA	12
3.1 Vaadin.....	13
3.2 IntelliJ IDEA	13
3.3 PAYARA.....	14
3.4 MySQL	14
3.5 Case studio 2.....	15
3.6 Jasper Reports.....	15
3.7 Visual Studio	16
4 POTEK DELA	17

5	OBLIKA SPLETNE APLIKACIJE	18
5.1	Začetna stran.....	18
5.2	Navigacija.....	19
5.3	Artikli	19
5.4	Naročilnice.....	20
5.5	Prevzemnice.....	23
5.6	Izdajnice.....	25
5.7	Dobavitelji	26
5.8	Stranke.....	27
5.9	Zaposleni	28
5.10	Skladišča	29
5.11	Lokacije v skladiščih.....	30
5.12	Artikli v skladiščih.....	31
6	KODA PROGRAMA.....	32
6.1	BL modul.....	33
6.2	DATA modul.....	34
6.2.1	Razredi entitet	35
6.2.2	Pojoti	36
6.2.3	Vmesniki (servisi)	36
6.2.4	Enumi	37
6.3	UI modul.....	37
6.3.1	Primarni razred za UI.....	38
6.3.2	Pogledi zavihkov	39
6.3.3	Forme na zavihkih (pogledih)	41
6.3.4	Dodatni specialni razredi za delovanje UI modula	42
7	ANKETA	43
7.1	Zaključki ankete	45
7.2	Ugotovitve glede na hipoteze	45
8	ZAKLJUČEK.....	47
9	VIRI IN LITERATURA	48
10	PRILOGE	49
10.1	Anketni vprašalnik	49

KAZALO SLIK

Slika 1: Shema skladiščnega poslovanja.....	10
Slika 2: Vaadin logotip.....	13
Slika 3: IntelliJ IDEA logotip.	14
Slika 4: Payara logotip.	14
Slika 5: MySQL logotip.	15
Slika 6: CASE Studio 2.....	15
Slika 7: Jaspersoft logotip.	16
Slika 8: Visual Studio logotip.	16
Slika 9: ER diagram baze.	18
Slika 10: Začetna stran.	18
Slika 11: Navigacija.	19
Slika 12: Zavihek »artikli«.....	19
Slika 13: Vpisani artikli.....	20
Slika 14: Urejanje artiklov.	20
Slika 15: Ustvarjanje naročilnice.	21
Slika 16: Artikli na naročilnici.	21
Slika 17: Seznam naročilnic.	22
Slika 18: PDF naročilnica.	22
Slika 19: Ustvarjanje prevzemnice.....	23
Slika 20: Artikli na prevzemnici.	23
Slika 21: Seznam prevzemnic.	24
Slika 22: Prevzemnica PDF.....	24
Slika 23: Ustvarjanje izdajnice.....	25
Slika 24: Artikli na izdajnici.	25
Slika 25: PDF izdajnica.....	26
Slika 26: Dodajanje dobaviteljev.	26
Slika 27: Seznam dobaviteljev.	27
Slika 28: Dodajanje strank.	27
Slika 29: Seznam strank.	28
Slika 30: Dodajanje zaposlenih.	28
Slika 31: Urejanje zaposlenih.	29
Slika 32: Dodajanje skladišč.	29
Slika 33: Seznam skladišč.	30

Slika 34: Dodajanje lokacij v skladišču.	30
Slika 35: Primer označevanja lokacij.	31
Slika 36: Dodajanje artiklov na lokacije.	31
Slika 37: Seznam artiklov v skladiščih.	32
Slika 38: Moduli projekta.	32
Slika 39: Razredi v modulu BL.	33
Slika 40: Primer metode v razredu control.	33
Slika 41: Razredi, enumi, pojoti in vmesniki na modulu data.	34
Slika 42: Ustvarjanje atributa v entiteti.	35
Slika 43: Konstruktor entitete.	35
Slika 44: Lastnosti atributa.	35
Slika 45: Konstruktor pojota naročilnice.	36
Slika 46: Vmesnik (servis).	36
Slika 47: Enum za enoto mase.	37
Slika 48: Razredi na modulu UI.	37
Slika 49: Metoda za spreminjanje porta in protokola ter sledenje seje.	38
Slika 50: Metoda za obnovitev seje.	38
Slika 51: Inicializacija komponent.	39
Slika 52: Metoda za vsak vstop.	39
Slika 53: Metoda za ob prvem vstopu.	40
Slika 54: Metoda za ustvarjanje forme.	41
Slika 55: Metoda za nastavljanje podatkov na formo.	41
Slika 56: Metoda za odpiranje PDF pregledovalnika.	42
Slika 57: Metoda za ustvarjanje in polnjenje PDF dokumenta.	42

KAZALO GRAFOV

Graf 1: Ste že kdaj uporabljali skladiščni sistem?	43
Graf 2: Ali veste, kaj je skladiščni sistem?	44
Graf 3: Ali se vam zdi skladiščni sistem uporaben?	44
Graf 4: Bi v svojem podjetju uporabili skladiščni sistem?	45

Povzetek

Glavni cilj raziskovalne naloge je bil ustvariti delujčo in enostavno spletno aplikacijo za pomoč v skladiščih.

Najin program za skladišča ima vse potrebne funkcije, za uporabo v skladišču. Vsebuje možnost naročanja, prevzemanja in izdajanja, hkrati pa lahko zraven vodimo evidenco, v katerem skladišču na kateri lokaciji je neka stvar, za katero velja, da je lahko ista stvar oziroma artikel na večih mestih in v večih skladiščih.

Projekt je pripravljen za uporabo v poslovne namene in je združljiv z ostalimi aplikacijami, ki jih uporablja neko podjetje. Program je napisan tako, da lahko kdor koli s potrebnim znanjem upravlja z njim, se pravi ga posodablja in tudi popravlja.

Abstract

Main goal of our task was to create operating and simple web application to help with work in warehouses.

Our program has all the functions needed in storage business. It contains function for ordering, takingover, issuing, at the same time we can manage informations about storage houses, on which location is some item which is considered to be the same item on many locations.

Project is ready to use in business purpose, and is compatible with other apps that any company can use. Program is written in that way that and with needed knowledge can manage it, update it and repair it.

Ključne besede

Skladiščni sistem, skladišče, Java EE, PAYARA, MySQL, IntelliJ IDEA, spletna aplikacija.

Keywords

Warehouse system, warehouse, Java EE, PAYARA, MySQL, IntelliJ IDEA, web application.

Kratice in okrajšave

- HTML (hyper text markup language),
- JavaScript (objektni skriptni programski jezik),
- EM (enota, mere),
- PDF (portable document format),
- BL (business logic),
- UI (user interface),
- POJO (plain old Java object).

1 UVOD

V zadnjih letih je vse več nakupov opravljenih preko interneta, saj želijo naročniki čim večjo ponudbo izdelkov, ki jih iščejo, splet pa jim to ponuja. Sorazmerno s prodajo preko spleta pa narašča tudi potreba po skladiščenju teh izdelkov in po njihovi organizaciji v skladišču. Tudi sama sva opazila ta trend, vendar na spletu nisva našla programa, ki bi ustrezal najnim zahtevam, zato sva se odločila, da ga napiševo sama.

Najprej sva se odločila, katero okolje in katere sisteme bova uporabljala in ali so kompatibilni. Ko sva ugotovila, da vse stvari delujejo, sva začela s pridobivanjem potrebnih znanj za najin projekt. Ko sva dosegla določeno raven znanja o programskem okolju in ostalih sistemih, sva začela z ustvarjanjem projekta, pri katerem nisva imela posebnih težav, saj sta nama pomagala podjetje BASS d.o.o. in najin mentor, sama pa sva tudi imela že določeno predznanje iz programiranja.

Projekt uporablja okolje Java EE in s pomočjo dodatka Vaadin ustvarjava kodo, napisano v Javi, zraven pa nama Vaadin ustvarja HTML5, CSS, JavaScript in PHP in s tem pripomore k lepši in bolj profesionalni obliki.

1.1 Predstavitev problema

Glavni problem, ki ga opisujeva v raziskovalni nalogi, je nepreglednost in neurejenost skladišč. Skladišče mora biti urejeno tako, da je prostor v njem optimalno izkoriščen. Vsi izdelki morajo biti zaradi večje učinkovitosti pri delu kar se da hitro dosegljivi in pregledno označeni. Največ težav je v velikih podjetjih, kjer sta obseg poslovanja in količina izdelkov velika. Prihaja do nepreglednost in nedostopnosti zahtevanih artiklov, saj se ti vedno odlagajo v skladišče, nihče pa ne vodi natančne evidence, kje stojijo in kam se odlagajo.

1.2 Hipoteze

Glede skladiščnih sistemov in njihove uporabe v vsakdanjem življenju, sva postavila nasledenje hipoteze. Te so:

- Hipoteza 1: večina anketiranih bo skladiščne sisteme označila kot učinkovite,
- Hipoteza 2: za izdelavo skladiščnega sistema bo potrebno izvenšolsko znanje,
- Hipoteza 3: uporaba skladiščnega sistema bo učinkovita in hitra,
- Hipoteza 4: skladiščni sistem ustreza vsem vrstam skladišč,
- Hipoteza 5: aplikacijo bo mogoče narediti brez stroškov.

1.3 Cilji

Kot cilj in namen raziskovalne naloge sva si zadala, da bova izdelala skladiščni sistem, ki bo hiter, učinkovit in pripravljen za uporabo v večjih podjetjih in skladiščih, poleg tega pa se bova uspešno naučila novih programskih jezikov in urejevalnikov.

1.4 Raziskovalne metode

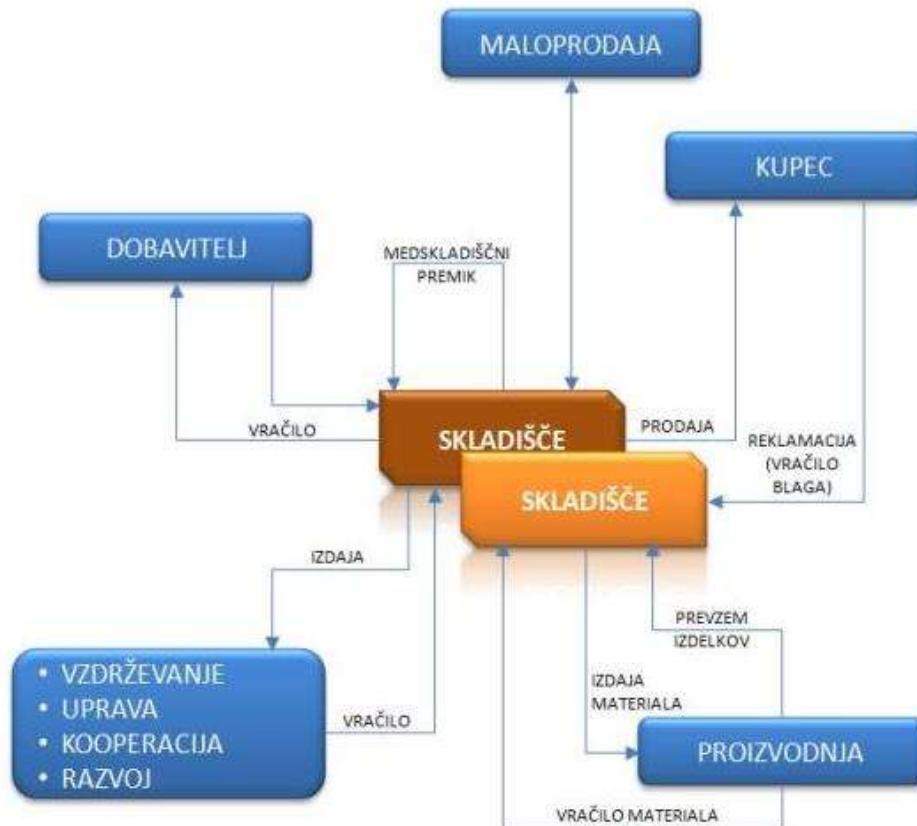
Delo sva si razdelila in med seboj uspešno sodelovala. Raziskala sva področje skladiščnih sistemov in njihovo ponudbo. Za preverjanje hipotez sva opravila kratko anketo med sošolci in v podjetju. Raziskala sva splet, da bi našla kakšne podobne primere, iskala pisne vire in se posvetovala z raznimi podjetji. Naloga je razdeljena na dva dela, in sicer na teoretičnega in praktičnega. V teoretičnem delu bova na kratko predstavila skladiščne sisteme in njihovo delitev, v praktičnem delu pa se bova osredotočila na najin program in ga tudi predstavila.

2 SKLADIŠČA IN DELITEV SKLADIŠČ

Skladišče je prostor, v katerem hrаниmo različne izdelke. Namenska poslovneža je imeti čim manj uskladiščenih izdelkov, saj to prinaša dodatne stroške. Ker čisto brez skladišč vseeno ne gre, je pametno imeti dobro izkoriščen prostor in dobro aplikacijo, ki ga upravlja, s tem pa lahko znižamo stroške na minimum.

Glavni vzroki za skladiščenje so:

- neskladnost med časom proizvodnje in časom uporabe izdelkov,
- oddaljenost med krajem proizvodnje in krajem potrošnje,
- različni trgovski in finančni pogoji,
- potreba po varnosti oskrbe.



Slika 1: Shema skladiščnega poslovanja.

Vir: <http://www.minoa.si/Default.aspx?ID=39>

2.1 Naloge skladišč

Osnovne naloge skladišč so prevzem in izdaja blaga ter varovanje blaga pred poškodbami in tatvinami. V času, ko je blago v skladišču morajo, delavci in skladišče samo zagotoviti, da izdelki in blago pri sprejemu, izdaji in shranjevanju ne bodo količinsko in kakovostno poškodovani. Naloge, ki sledijo prej omenjenima, pa so tudi:

- prerazporejanje blaga,
- kontrola stanja zalog,
- skrb za red in čistočo,
- optimalna razporeditev blaga,
- ekonomično poslovanje.

2.2 Vrste skladišč glede na način gradnje

Različne vrste skladišč omogočajo shranjevanje in varovanje različnih izdelkov glede na gradnjo, varovanje in pogoje obratovanja.

2.2.1 Odprto skladišče

Ko govorimo o odprttem skladišču, imamo v mislih skladišče brez strehe. Ta vrsta je primerna za blago in izdelke, ki niso občutljivi na različne zunanje in vremenske vplive (pesek, robniki, tlakovci ...). Dovazi morajo biti utrjeni in tlakovani, saj se takšna skladišča največkrat uporablja za težke surovine in gradbeno opremo. Prostor kjer se nahajajo mora biti suh in primerno vzdrževan.

2.2.2 Pokrito skladišče

Pokrito skladišče je tisto, ki je pokrito s streho. Uporablja se za shranjevanje težkega in robustnega blaga, na katerega veter, temperaturne razlike, vlaga in dež ne vplivajo. Največkrat je tam skladiščeno blago manjše vrednosti.

2.2.3 Zaprto skladišče

V zaprtem skladišču na blago vremenski vplivi in pojavi nimajo učinka. Največkrat se v njih shranjuje blago, ki zahteva določene razmere za skladiščenje. Praviloma so skladišča te vrste razdeljena na oddelke, glede na vrsto blaga. Takšni prostori so po navadi zidani, zaklenjeni in varovani, saj vsebujejo blago večje vrednosti kot pri odprttem in pokritem skladišču.

2.2.4 Specialno skladišče

Specialna skladišča se uporablja za varovanje blaga velike vrednosti ali za blago, ki je v normalnih pogojih nevarno. Sem spadajo trezorji (varovanje zelo vrednega materiala), silosi, vinske kleti, skladišča za eksplozivne ali vnetljive materiale (varovanje blaga, ki mora biti zaščiteno pred posebnimi zunanjimi vplivi).

2.3 Vrste skladišč glede na vrsto blaga

Zaradi različnih vrst blaga in njihovega načina shranjevanja so se razvile različne vrste skladišč:

- glavno skladišče, ki sprejme in zabeleži vso blago, ki je namenjen skladiščenju in odprenji
- pomožno skladišče, ki se uporablja kot pomožno, če je glavno prezasedeno
- priročno skladišče, ki je namenjeno proizvodnim in trgovinskim mrežam, uporabljeno zaradi hitrejše dostave in oskrbe poslovalnic
- specialno skladišče, ki je namenjeno blagu, ki zahteva posebne pogoje skladiščenja in

- sprejemno skladišče, v katerem je blago le toliko časa, dokler se ne odpremi v glavno skladišče.

2.4 Glavni skladiščni procesi

V vsakem skladišču je delo drugačno, skladiščni procesi pa so povsod enaki. Njihov namen je zagotoviti čim večjo uspešnost, ki je odvisna od organizacije dela, preglednosti, enostavnosti, zagotavljanja čim krajsih transportnih poti in čim manj premeščanja blaga po skladišču. Sprejeto blago mora biti skladiščeno tako, da je pregledno in dosegljivo za vsakogar in ga je lahko nadzirati.

2.4.1 Prevzem blaga

Prevzem blaga je točka, kjer blago prvič pride v skladišče. Prvi korak je pregled dokumentacije ter preverjanje količine, ki mora biti enaka tisti na naročilnici in dobavnici. Sledi kratek pregled kakovosti, pri vsem tem pa je treba paziti, da se blago ne poškoduje.

2.4.2 Skladiščenje blaga

Pri skladiščenju blaga se opravi natančnejši kakovostni pregled, s katerim se zagotovi, da prispelo blago ni fizično in količinsko poškodovano. Pregledano blago se nato razvrsti po skladišču, kjer ostane do odpreme.

2.4.3 Odprema blaga

Pri odpremi se blago ustrezno pripravi ali zapakira, vsi podatki o odpremi so zapisani na izdajnici, s katero delavec poišče izdelke in jih pripravi za kupca.

2.4.4 Komisioniranje blaga

Komisioniranje je skladiščni proces, ki se opravlja med skladiščenjem in odpremo izbranega blaga. To je delo, ko se blago zbira po skladišču glede na zahteve na naročilnici in se nato dostavi na zbirno mesto, kjer se preveri številčna ustreznost in se nato zapakira in pripravi na odpremo.

3 PROGRAMSKA OPREMA

Pri izdelavi najinega izdelka sva uporabila veliko različnih računalniških orodij in področij, ki so na spletu na voljo vsakemu in so brezplačni. Pomemben podatek, ki bi ga izpostavila, je da

sva se vseh teh orodji, poleg Visual studia in Case studia 2, naučila sama, kar zaradi predznanja o teh jezikih ni bil problem. Najpomembnejša orodja so opisana v nadaljevanju.

3.1 Vaadin

Vaadin je odprtokodna platforma za razvoj in uporabo spletnih aplikacij. Namenjen je enostavenmu razvoju, vzdrževanju, uporabi in razširljivosti. V uporabi je od leta 2001 in velja za enega najbolj uporabnih pri razvoju in izdelavi zahtevnejših spletnih aplikacij.

Namen Vaadina je, da uporabnik ali razvijalec porabi čim manj časa za učenje in osvajanje spletnih tehnologij in se lahko zato osredotoči na čim večjo uporabnost. Ne zahteva znanja HTML ali skriptnega jezika JavaScript kot nekateri drugi. Uporabniški vmesniki se izvajajo kot JavaScript, zato dodatni spletni vtičniki niso potrebni.



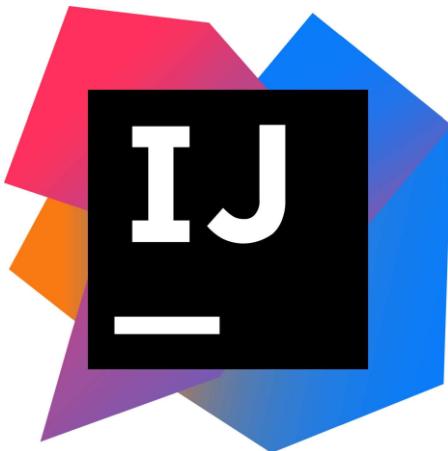
Slika 2: Vaadin logotip.

Vir: [Vaadin Logo](#)

3.2 IntelliJ IDEA

IntelliJ IDEA je razvojno okolje za Java Razvijalce programske opreme. Začetki segajo v leto 2001, ko je bil to prvi urejevalnik, ki je ponujal avtomatično reorganizacijo napisane kode. IntelliJ temelji na hitrem programiranju, ki razvijalcu omogoča osredotočanje na razvoj kode in ne na sintaktično pravilnost. S tem namenom ima IntelliJ vgrajene različne mehanizme, ki avtomatsko preverjajo ustreznost napisane kode, komentarjev, samodokončujejo napisano kodo in predlagajo ustrezne rešitve za morebitne napake, ki se pojavit ob pisanju kode. Zanimiv je tudi zato, ker podpira večino programskih jezikov, ki smo se jih učili v šoli, in sicer:

- JavaScript,
- HTML,
- CSS,
- SQL,
- PHP

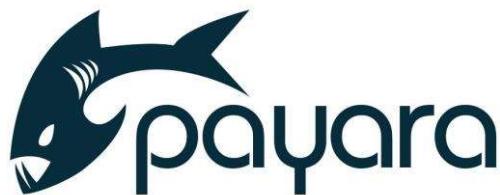


Slika 3: IntelliJ IDEA logotip.

Vir: [IntelliJ IDEA Logo.svg](#)

3.3 PAYARA

Payara Server je odprtokodni aplikacijski server in je naslednik bolj poznanega GlassFish Serverja. Ustvarjen je bil leta 2014 z namenom zamenjave GlassFish Serverja podjetja Oracle. Payara Server je odprtokodna vmesna platforma, ki podpira zanesljive in varne postavitve aplikacij Java EE (Jakarta EE) v vsakem okolju: v prostoru, oblaku ali hibridnem okolju.



Slika 4: Payara logotip.

Vir: [Payara Server Logo](#)

3.4 MySQL

MySQL je odprtokodni sistem za upravljanje s podatkovnimi bazami. Za delo uporablja jezik SQL. Napisan je v programskih jezikih C in C++. Temelji na modelu odjemalec-strežnik, torej gre za povezavo vsaj dveh računalnikov. Pri tem strežnik ponuja podatke ali storitve, odjemalec pa se preko omrežja povezuje s strežnikom in uporablja ponujene podatke ali storitve. Praviloma se uporablja en strežnik in več odjemalcev. MySQL je bil primarno razvit za hitro zapisovanje velikih podatkovnih baz. Uporabnikom MySQL se ni potrebno učiti

novih ukazov, saj lahko dostopajo do svojih podatkov s standardnimi ukazi SQL in jih spreminjajo. Prednost MySQL je tudi uporaba na več platformah, saj podpira Windows, Linux in Mac OS.

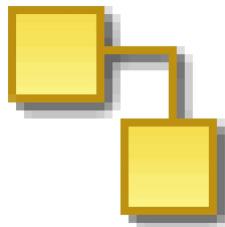


Slika 5: MySQL logotip.

Vir: [MySQL Logo](#)

3.5 Case studio 2

CASE Studio 2 je profesionalno orodje za načrtovanje podatkovne baze, ki omogoča vizualno izdelavo diagramov entitetnih odnosov (ERD) za različne sisteme baz podatkov - npr. MS SQL Server, Oracle, DB2, Access, Sybase, Firebird, MaxDB, MySQL, PostgreSQL in drugo. S CASE Studio 2 lahko zlahka obrnete inženirske strukture že obstoječih podatkovnih baz. Funkcije vključujejo generiranje SQL skriptov, upravljalnik različic, uporabniško določene predloge in poročila HTML.



Slika 6: CASE Studio 2.

Vir: [Case Studio 2 Logo](#)

3.6 Jasper Reports

Jasper Reports je odprtakodno orodje za poročanje. Pogost je pri uporabi aplikacij, ki omogočajo Java, Java EE in aplikacij za ustvarjanje dinamične vsebine. Je eden najbolj uporabnih orodij saj lahko podatke prikaže kot:

- PDF,
- HTML,

- XML ali
- Microsoft Excel.

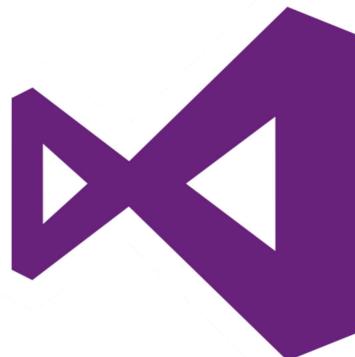


Slika 7: Jaspersoft logotip.

Vir: [Jaspersoft Logo](#)

3.7 Visual Studio

Microsoft Visual Studio je razvojno okolje podjetja Microsoft, namenjeno razvoju za Windows OS, spletnih strani, aplikacij in storitev na podlagi ogrodja .NET. Okolje je zelo uporabno, saj lahko programira tako upravljano, kot tudi strojno kodo, kar je neke vrste posebnost. Možna je tudi implementacija drugotnih urejevalnikov, eden od teh so storitve podjetja Xamarin. Urejevalnik je zelo uporabniku prijazen, saj kodo dokončuje in dopolnjuje sam. Brezplačno si je možno namestiti verzijo Visual Studio Express, vendar ta ne podpira vtičnikov in drugotnih storitev drugi neodvisnih razvijalcev. Za uporabo vseh vtičnikov je na voljo Visual Studio Community Edition, ki je prav tako brezplačen.



Slika 8: Visual Studio logotip.

Vir: [Visual Studio Logo](#)

4 POTEK DELA

Začela sva z idejo o spletni aplikaciji za skladišča, saj nisva našla takšne, ki bi ustrezała najinemu okusu. Uporabila sva veliko novih orodji, katerih funkcije sva morala najprej osvojiti in kasneje tudi uporabiti, pri čemu nama je zelo pomagalo podjetje BASS d.o.o.. Ko sva osvojila potrebno znanje za najino idejo, sva pričela z izdelavo. Svoj projekt sva razdelila na tri dele: UI (user interface – uporabniški vmesnik), data (baza, konstruktorji in lastnosti) in BL (business logic – poslovna logika ali metode za manipuliranje z bazo in podatki).

Najprej sva po posvetu z mentorjem ustvarila bazo, ki sva jo skozi ustvarjanje tudi optimizirala in spreminjala. Ko je bila baza pripravljena za uporabo, sva pričela s postavljanjem okolja vseh potrebnih stvari za delovanje. Ko sva uspešno postavila bazo in jo pognala, hkrati pa tudi spletni strežnik za delovanje spletne strani, sva začela s programiranjem.

Najprej sva morala ustvariti vse entitete, njihove atribute, konstruktorje in lastnosti, saj sva uporabila razredno ustvarjanje oziroma implementacijo baze.

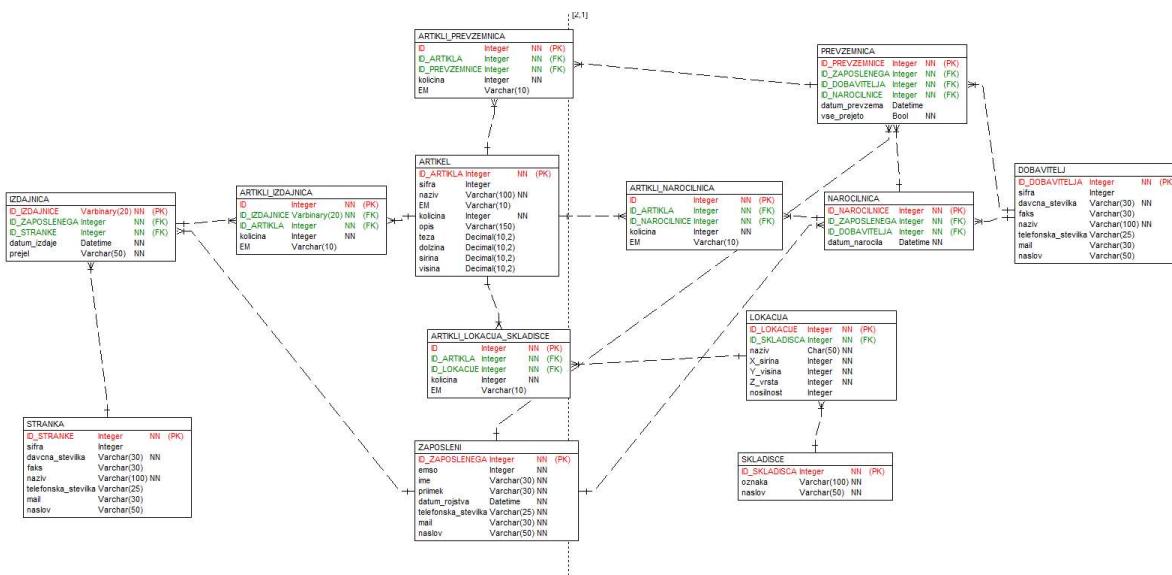
Ko je bila baza dokončno implementirana sva začela s skiciranjem vseh zavihkov oziroma oblike strani na splošno. Ko sva oblikovala izgled, primeren za uporabo, sva pričela z ustvarjanjem vmesnikov za te entitete in kontrolnih razredov, ki vsebujejo metode za manipuliranje z entitetami.

Po tem, ko sva ustvarila prvi in drugi del najinega projekta, to sta BL in data modul, sva pričela z najtežjim to je UI modul. Najprej sva ustvarila začetno stran, nato pa vse ostale zavihke, ki delujejo po istem principu. Ko odpreno nek zavihek, se nam pokažejo vsi podatki, ki jih lahko urejamo ali ustvarimo nove. Glavna funkcija, za katero se nama zdi, da bi jo morala posebej omeniti, pa je priprava PDF dokumentov, ki se požene le s klikom na gumb. Pri programiraju UI modula nama je pomagalo podjetje BASS d.o.o., ker so nama svetovali glede poteka dela in delovanja skladiščnega sistema.

5 OBLIKA SPLETNE APLIKACIJE

Vsaka stran pri spletni aplikaciji ima svoj namen, ustvarjen pa mora biti tako, da je uporaba kar se da enostavna.

Najprej sva v programu CASE Studio 2 ustvarila ER diagram celotne baze, kasneje pa sva jo ustvarila s pomočjo razrednega programiranja.



Slika 9: ER diagram baze.

5.1 Začetna stran

Začetna stran vsebuje samo najpomembnejše podatke, ki jih potrebuje tisti, ki uporablja spletno aplikacijo. Vidno je število naročilnic, prevzemnic, izdajnic in zalog izdelkov, ki so zbrani v skladišču.

The screenshot shows the following statistics:

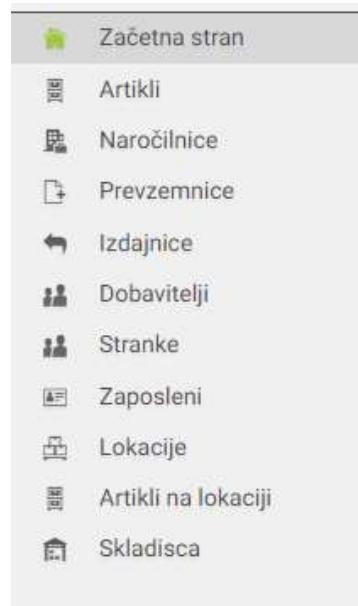
- Število naročilnic: 15
- Število prevzemnic: 8
- Število izdajnic: 7

Sifra	Naziv	Kolicina	EM
C01	Cev B	5	KOS

Slika 10: Začetna stran.

5.2 Navigacija

Navigacija spletne aplikacije za skladiščenje je enostavna in minimalistična, vsebuje povezave do zavihkov, ki so potrebni za uporabo celotne aplikacije.



Slika 11: Navigacija.

5.3 Artikli

Prvi zavihek ima funkcijo dodajanja artiklov, ki prispejo v skladišče. Po kliku na gumb za dodajanje se odpre okno, v katerega lahko vpišemo vse potrebne podatke o prispelem artiklu.

The screenshot shows the 'Artikli' (Articles) section of the application. On the left is a sidebar with navigation links. The main area displays a table of articles with columns: Interna šifra (Internal code), Naziv (Name), Količina (Quantity), EM (Unit), Dolžina[MM] (Length [MM]), Širina[MM] (Width [MM]), Višina[MM] (Height [MM]), Teža[KG] (Weight [KG]), and Datum veljavnosti (Validity date). Two rows are visible: 'Cev B' with code 'C01' and 'Vijak' with code 'V01'. A modal window titled 'Uredanje / dodajanje' (Editing / Adding) is open in the center. It contains fields for 'Interni šifra:' (Internal code), 'Naziv:' (Name), 'Znamka:' (Brand), 'Količina:' (Quantity), 'EM:' (Unit), 'Datum veljavnosti:' (Validity date), 'Teža[KG]:' (Weight [KG]), 'Dolžina[MM]:' (Length [MM]), 'Širina[MM]:' (Width [MM]), and 'Višina[MM]:' (Height [MM]). At the bottom of the modal are buttons for 'štorev' (Save) and 'Prekliči' (Cancel).

Slika 12: Zavihek »artikli«.

Seznam vpisanih izdelkov se samodejno osvežuje z vsakim klikom na gumb shrani. Nad seznamom izdelkov je tudi vrstica za iskanje. Iščemo lahko po nazivu ali šifri izdelka.

Skladiščni sistem CP									
Artiki									
	Vrsta iskanja	Iskanje...			Počisti		Dodaj artikel		
	Interni šifra	Naziv	Količina	EM	Dolžina[MM]	Širina[MM]	Višina[MM]	Teža[KG]	Datum veljavnosti
	C01	Cev B	5	KOS	2000	20	20	20	
	V01	Vijak	1000	KOS	70	3	3	0	
	PVC001	PVC cev 16"	100	KOS	5000	16	0	1	
	PVC10x10	PVC kanal 10x10	500	KOS	2000	10	10	0	
	PVC17x17	PVC kanal 17x17	500	KOS	2000	17	17	0	
	PVC40x40	PVC kanal 40x40	500	KOS	2000	40	40	0	
	NZB001	Jupol classic BELA	250	KOS	0	0	0	16	31.03.2029
	NZB002	Jupol classic RUMENA	400	KOS	0	0	0	5	29.03.2030
	C0001	Ploščati čopici 3/1	150	ZABOJ	0	0	0	0	
	C0002	Ploščati čopici	3000	KOS	0	0	0	0	
	TG001	Guma 385/65R 22.5	100	KOS	0	0	0	18	
	TG002	Guma 295/80R 22.5	75	KOS	0	0	0	20	
	TG003	Guma 315/80R 22.5	100	KOS	0	0	0	23	
	AP001	Viljamovka 30%	500	KOS	0	0	0	1	09.03.2024
	AP002	Viski Balantines 40 vol%	500	KOS	0	0	0	1	25.03.2032
	AP003	Čokoladni liker 18 vol%	500	KOS	0	0	0	1	18.12.2025
	AP004	Rum domači 40 vol%	500	KOS	0	0	0	1	29.07.2038
	AP005	Borovnikevec 20 vol%	500	KOS	0	0	0	1	18.03.2022
	AP006	Viski Jameson 40 vol%	500	KOS	0	0	0	1	24.02.2028
	BAP001	Nektar jabolka	2000	KOS	0	0	0	1	05.03.2020
	BAP002	Nektar breskev	2000	KOS	0	0	0	1	08.04.2020
	BAP003	Nektar jagoda	2000	KOS	0	0	0	1	27.03.2020
	PPV002	Protipoplavne vrčeve 20kg	10	PALETA	400	300	100	20	
	PPV001	Protipoplavne vrčeve 50kg	15	PALETA	800	600	150	50	
	SP001	Prikupljena spojka 2273	1500	KOS	0	0	0	0	
	SP002	Povezovalna sponka	500	KOS	15	5	2	0	

Slika 13: Vpisani artikli.

Za spremjanje podatkov pri že vnesenih artiklih samo kliknemo nanj in spremenimo vrednosti v oknu, ki se nam prikaže, ter kliknemo »shrani«. Podatke in informacije, ki so za vpis izdelka nujno potrebni sva označila z rdečo zvezdico (*), da je vse čim bolj pregledno.

Urejanje / dodajanje

Interni šifra:	C01	Datum veljavnosti:	Preklici
Naziv: *	Cev B	Teža[KG]:	20,00
Znamka: *	ROSENBOUER	Dolžina[MM]:	2000,00
Količina: *	9	Širina[MM]:	20,00
EM:	KOS	Višina[MM]:	20,00

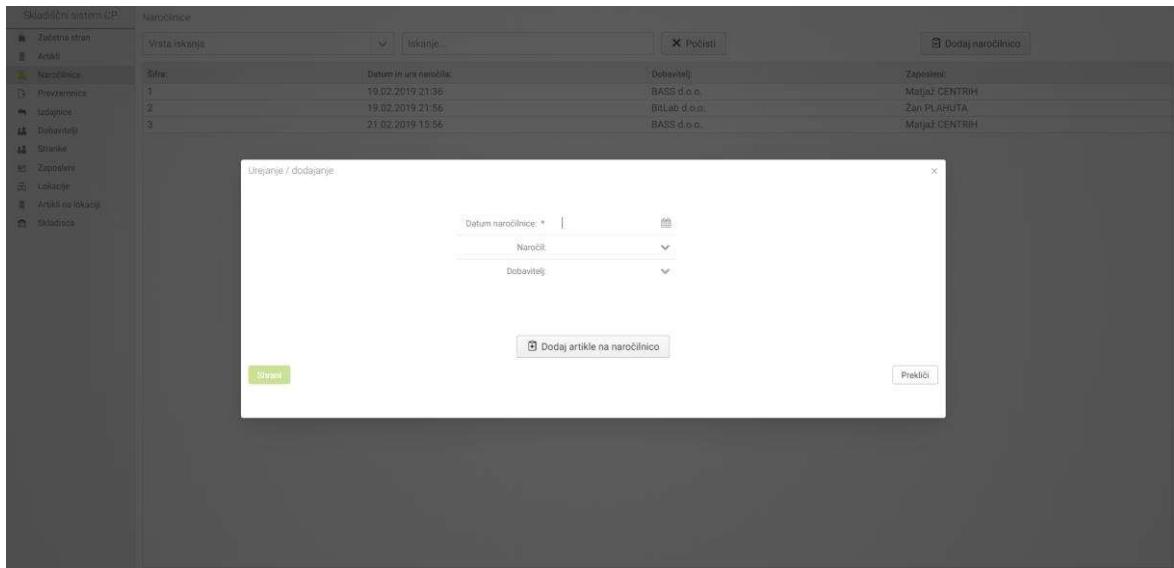
Shrani

Slika 14: Urejanje artiklov.

5.4 Naročilnice

Za dodajanje nove naročilnice enostavno kliknemo na gumb »dodaj naročilnico in odpre se nam novo okno, na katerega vpišemo datum naročilnice, ime naročnika ter ime dobavitelja. Dobavitelja najdemo na seznamu in ga ni potrebno vpisovati ročno. Aplikacija podatke o

dobaviteljih pridobi iz »data providerja«, ki v bazo pošlje zahtevo po imenih dobaviteljev in nam jih prikaže v seznamu.



Slika 15: Ustvarjanje naročilnice.

Ko je naročilnica ustvarjena, lahko začnemo z dodajanjem artiklov na seznam. Artikli in EM se prav tako kot imena dobaviteljev prenesejo na seznam z »data providerja«, količino pa moramo vpisati sami.



Slika 16: Artikli na naročilnici.

Po končanem dodajanju in urejanju naročilnic se nam na zavihku prikažejo vse naročilnic, ki smo jih ustvarili.

Skladiščni sistem CP					
Naročilnice					
	Vrsta iskanja	Iskanje...	X Počisti	Dodaj naročilnico	Natisni
Šifra:	Datum in ura naročila:	Dobavitelj:	Zaposleni:	Natiani	
1	19.02.2019 21:36	BASS d.o.o.	Matjaž CENTRIH		
2	19.02.2019 21:56	BitLab d.o.o.	Žan PLAHUTA		
3	21.02.2019 15:56	BASS d.o.o.	Matjaž CENTRIH		
4	04.03.2019 08:14	BASS d.o.o.	Matjaž CENTRIH		
5	04.03.2019 09:00	BASS d.o.o.	Žan PLAHUTA		
6	06.03.2019 19:38	BASS d.o.o.	Žan PLAHUTA		
7	08.03.2019 19:38	BitLab d.o.o.	Mateja ZVEGLER		
8	05.03.2019 19:38	Krka d.d.	Gáspér FUNDA POVŠE		
9	27.02.2019 19:39	Pipistrel d.o.o.	Rocky BRLIČ		
10	23.03.2019 19:39	BitLab d.o.o.	Mateja ZVEGLER		
11	26.03.2019 19:39	Slovenske železnice d.o.o.	Nejc RIHTER		
12	20.03.2019 19:39	Mercator d.d.	Nejc RIHTER		
13	07.03.2019 19:39	Alples d.d.	Nina KOSEM		
14	21.03.2019 19:40	Pipistrel d.o.o.	Max LAZNIK		
15	21.03.2019 19:40	Akrapovič d.d.	Kaja ČUK		

Slika 17: Seznam naročilnic.

Ob kliku na ikono »natisni« se nam na zaslonu prikaže PDF oblika naročilnice, ki jo lahko natisnemo in uporabimo ob dostavi naročila.

Naročilnica

 Številka naročilnice: 4

Datum in ura naročilnice: 04.03.2019 08:14:00 Št.strani: 1
Datum tiskanja: 06.03.2019

Podatki o zaposlenemu:

IME:	PRIIMEK:	MAIL:	TELEFON:
Matjaž	CENTRIH	matlazm@centrih.si	+386 051648133

Podatki o dobavitelju:

NAZIV:	DAVNA:	MAIL:	TELEFON:
BASS d.o.o.	124235	podpora@bass.si	+386 03456789

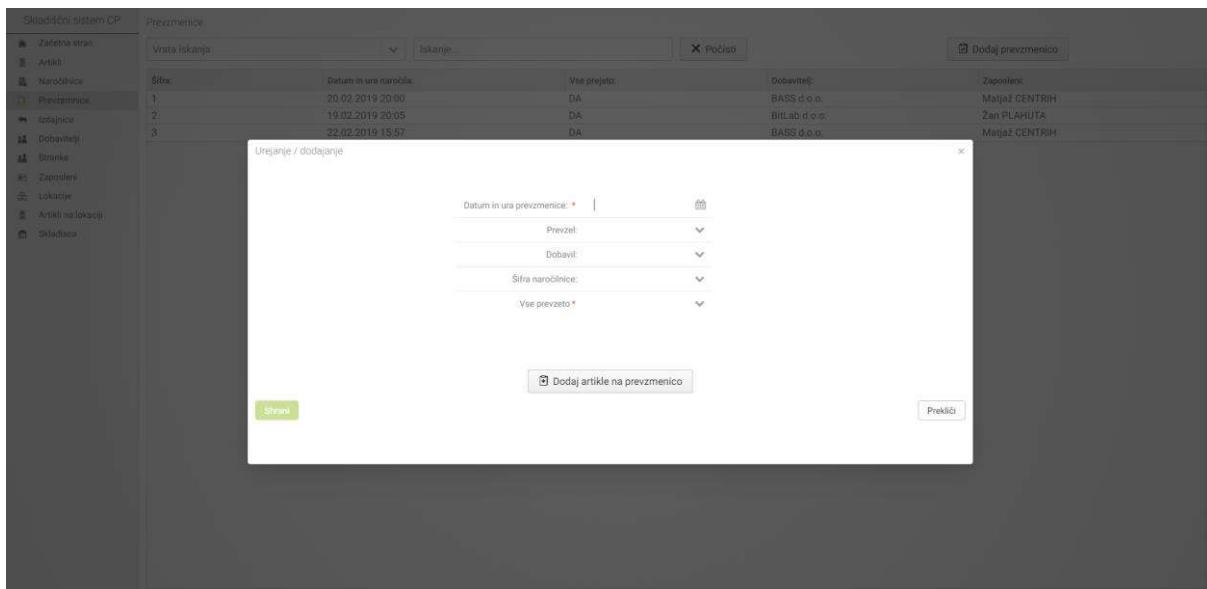
Artikli na naročilnici:

Naziv artikla	Kolina	EM
Cev B	50	KOS
Vijak	20	KOS
Vijak	20	ZABOJ
Cev B	2	KOS
Cev B	12	KOS
Vijak	12	KOS
Cev B	12	KOS
Cev B	34	ZABOJ
Cev B	56	KOS
Cev B	34	KOS

Slika 18: PDF naročilnica.

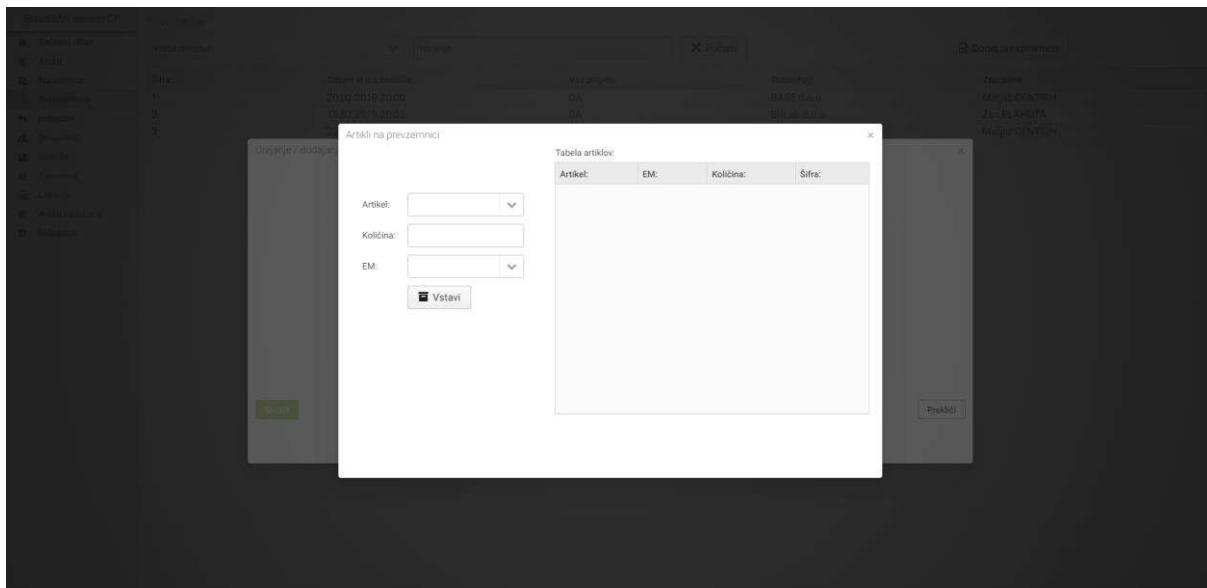
5.5 Prevzemnice

Prevzemnice ustvarjamo z gumbom »dodaj prevzemnico«. Prikaže se okno, v katerega vpišemo datum prevzema, ime osebe, ki bo blago prevzela, dobavitelja, šifro naročilnice, povezane z izdajnico, in kot zadnje, ali je bilo celotno naročilo že prevzeto ali ne.



Slika 19: Ustvarjanje prevzemnice.

Artikle, ki jih želimo imeti na prevzemnici, izberemo iz spustnega seznama, količino vpišemo sami, pod EM pa izberemo, ali je blago kosovno, na paleti ali v zaboju.



Slika 20: Artikli na prevzemnici.

Na glavnem pogledu zavihka prevzemnice so vidne vse ustvarjene prevzemnice. Vsako lahko tudi natisnemo.

Prevzemnice						
Vrsta iskanja	Iskanje...	X Počisti	Dodaj prevzemanico	Natisni		
Sifra:	Datum in ura naročila:	Vse prejeti:	Dobavitelj:	Zaposleni:		
1	20.02.2019 20:00	DA	BASS d.o.o.	Matjaž CENTRIH		
2	19.02.2019 20:05	DA	BitLab d.o.o.	Žan PLAHUTA		
3	22.02.2019 15:57	NE	BASS d.o.o.	Matjaž CENTRIH		
4	06.03.2019 19:41	DA	BitLab d.o.o.	Matjaž CENTRIH		
5	16.03.2019 19:42	NE	BitLab d.o.o.	Mateja ZVEGLER		
6	04.03.2019 19:42	NE	Telekom Slovenije d.d.	Nejc RIHTER		
7	20.03.2019 19:43	DA	Akrapovič d.d.	Gašper FUNDA POVŠE		
8	26.03.2019 19:43	DA	GRATEL d.o.o.	Nejc RIHTER		

Slika 21: Seznam prevzemnic.

PDF oblika prevzemnice vsebuje vse ključne podatke, ki so potrebni pri prevzemu naročenih artiklov.

Prevzemnica

CPSYSTEMS

Številka prevzemnice: 2

Datum in ura prevzemnice:	19.02.2019 08:05:00	Št.strani:	1
Vse prejeti:	DA	Datum tiskanja:	06.03.2019

Podatki o zaposlenemu:

IME:	PRIIMEK:	MAIL:	TELEFON:
Žan	PLAHUTA	zan.plahuta@gmail.com	+386 070123589

Podatki o dobavitelju:

NAZIV:	DAVNA:	MAIL:	TELEFON:
BitLab d.o.o.	846490	podpora@bitlab.si	+386 05789456

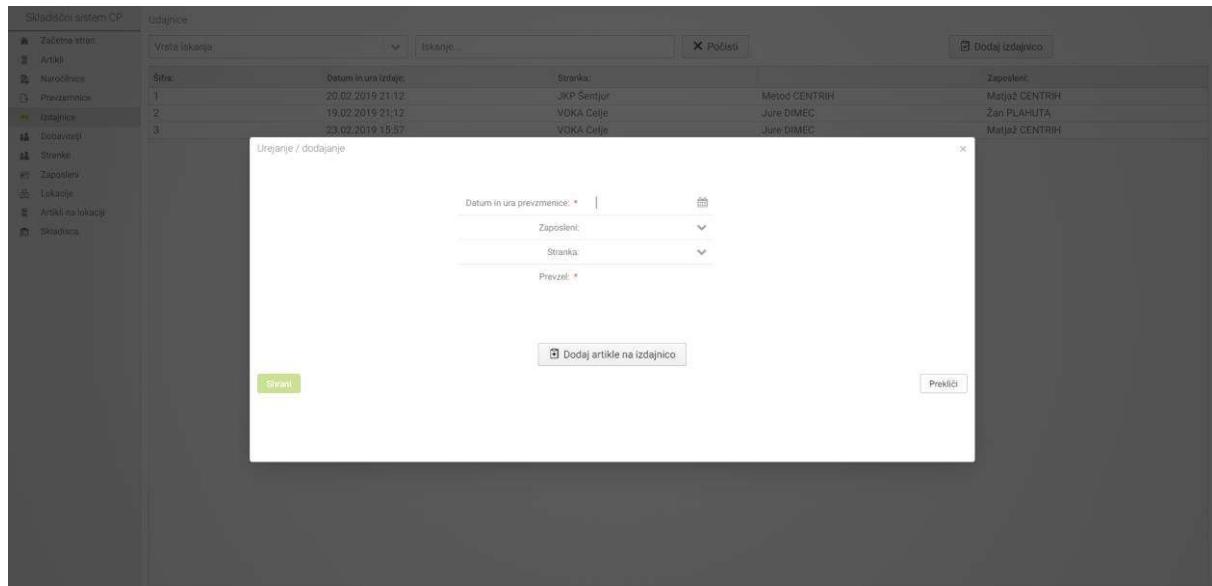
Artikli na prevzemnici:

Naziv artika	Kolina	EM
Vijak	500	KOS
PVC kanal 10x10	5	KOS
Cev B	10	KOS

Slika 22: Prevzemnica PDF.

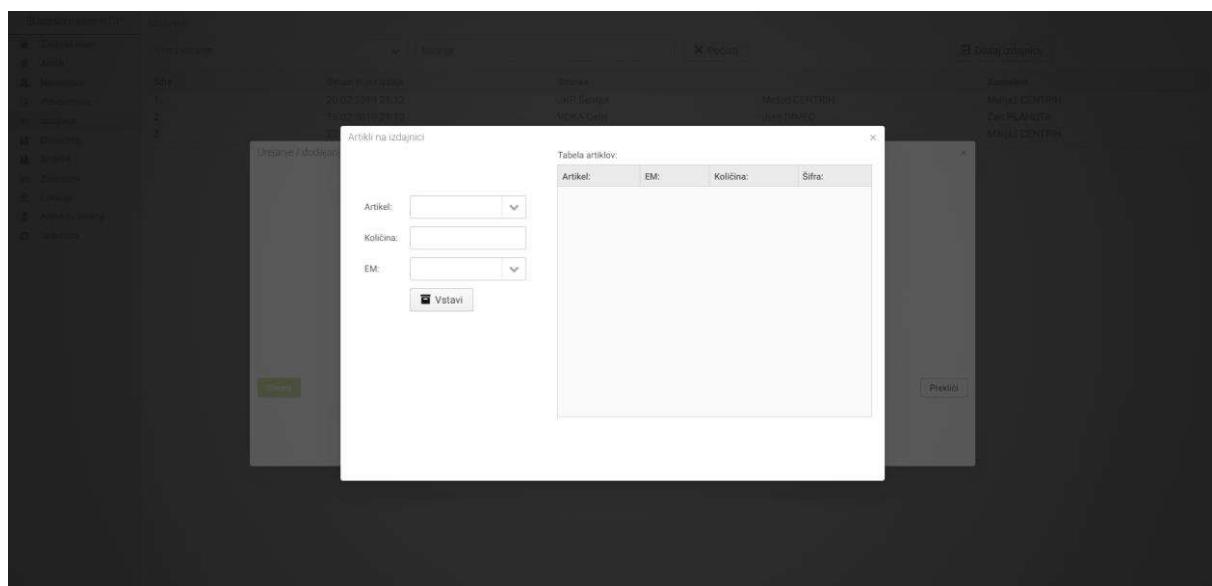
5.6 Izdajnice

Tudi izdajnico ustvarimo enako kot naročilnico in prevzemnico s klikom na gumb »ustvari izdajnico«. Izberemo datum tistega dne, ko smo izdajnico ustvarili, ime zaposlenega, ime stranke in ime tistega, ki bo blago prevzel.



Slika 23: Ustvarjanje izdajnice.

Artikle dodajamo podobno kot pri naročilnici in prevzemnici iz seznama izberemo artikel, mu določimo količino in izberemo ustrezne EM.



Slika 24: Artikli na izdajnici.

Tudi izdajnica ima svojo obliko pripravljeno za tiskanje, na njej je označena številka izdajnice, datum in ura, ko je bila izdajnica ustvarjena, število strani, ime prevzemnika, datum izvršitve, podatki o zaposlenem, ki je ustvaril izdajnico, in stranki, ki je artikle naročila, ter vse artikle, ki so vključeni na izdajnico.

Naziv artika	Kolina	EM
Vijak	500	KOS
PVC cev 16"	10	KOS
Protiplipalne vree 20kg	2	PALETA

Slika 25: PDF izdajnica.

5.7 Dobavitelji

Ker lahko ima skladišče več dobaviteljev, je potrebno vsakega novega vpisati. To opravimo s pritiskom na gumb »dodaj dobavitelja«. Za novi vpis je potrebno imeti njihov naziv, davčno številko podjetja, naslov, kjer poslujejo, faks, telefonsko številko in elektronski naslov.

Slika 26: Dodajanje dobaviteljev.

Na seznamu se prikažejo vsi že vstavljeni dobavitelji. Njihove podatke lahko spremojamo s klikom. Po kliku se odpre okno, enako tistemu kot pri dodajanju, kjer lahko podatke spremenimo in ponovno shranimo.

Slika 27: Seznam dobaviteljev.

5.8 Stranke

Podatki, potrebni za dodajanje strank, so njihov naziv, davčna številka, naslov, faks, telefonska številka in elektronski naslov. Po vpisu jih lahko shranimo in stranka bo dodana na obstoječi seznam.

Slika 28: Dodajanje strank.

Na seznamu lahko vidimo vse že vpisane stranke. Urejamo in spremojamo jih s klikom na tisto, kar želimo spremeniti.

Stranke					
	Vrsta iskanja	Iskanje...	X Počisti	Dodaj stranko	
	Naziv:	Dnevna številka:	Naslov:	Faks:	Mail:
	JKP Šentjur	489658	Cesta Leona Dobrotnička 11	46854612	podpora@jkpsent.si
	VOKA Celje	348564	Mariborska cesta 105	49658123	podpora@vokac.si
	Bahami s.p.	654132	Ljubljanska cesta 54	75836412	podpora@bahami.si
	Vilkograd d.o.o.	783453	Ulica Franja Malgaja 9	78932123	podpora@vilkograd.si
	Prevozništvo Maček	2113545	Kosovelova 166	87453632	podpora@macelj.si
	Gradnje Saramati d.o.o.	786745	Hruševci 12	32156846	podpora@saramati.si
	Jager d.d.	425647	Planina 22	75464237	podpora@jager.si
	KEA d.o.o.	654123	Malgajeva 77	98741321	podpora@kea.si
	Elektro Selic d.o.o.	654213	Krajska ulica 4	12343211	podpora@selic.si
	Polje d.o.o.	984162	Pekin 65c	32756497	podpora@polje.si
	Avrigo d.o.o.	315321	Pot na lom 121	72437514	podpora@avrigo.si
	Merkur d.d.	685730	Dolga vas 1	76542300	podpora@merkur.si
	Prevozi Mangart	584600	Pod Mangartom 3	09178413	podpora@mangart.si
	Kmetijska zadruga Šentjur	695111	Cesta Frana Roša 5	09172423	podpora@kms.si

Slika 29: Seznam strank.

5.9 Zaposleni

Vsek zaposleni v skladišču mora posredovati podatke o svojem imenu in priimku, EMŠO, datum rojstva, stalno ali začasno prebivališču, telefonsko številko in elektronski naslov. Vse te podatke dodamo z gumbom »dodaj zaposlenega«.

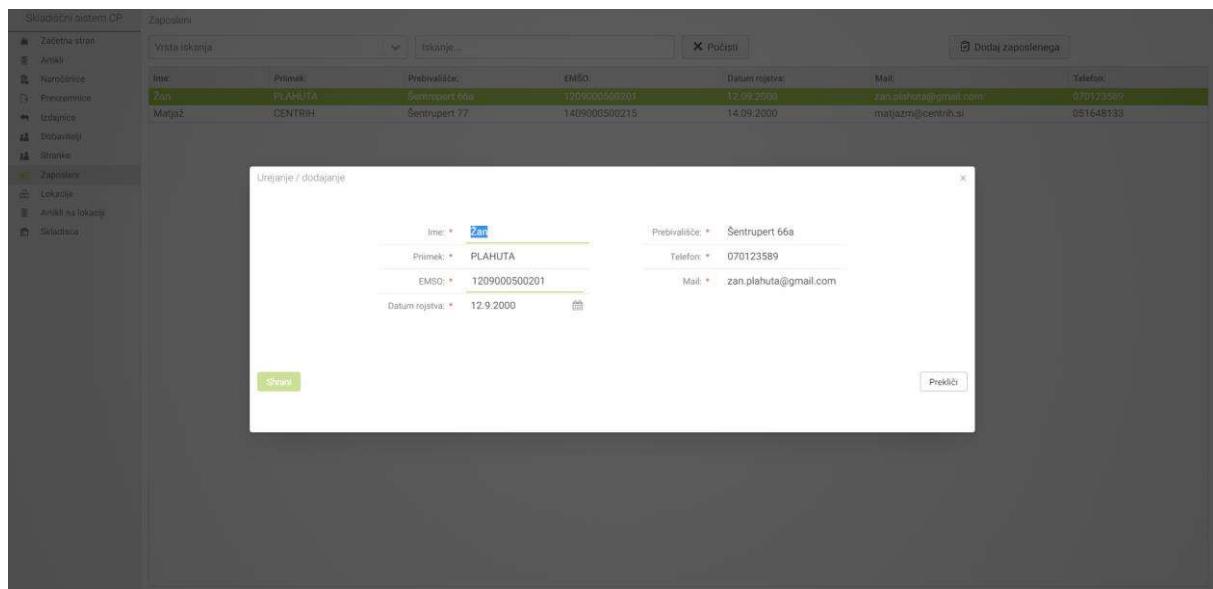
Zaposleni						
	Vrsta iskanja	Iskanje...	X Počisti	Dodaj zaposlenega		
	Ime:	Primék:	Prebivališče:	EMŠO:	Datum rojstva:	Mail:
	Zan	PLAHUTA	Šentrupert 66a	1209000500201	12.09.2000	zan.plahuta@gmail.com
	Matjaž	CENTRIH	Šentrupert 77	1409000500215	14.09.2000	matjazm@centrih.si

Urejanje / dodajanje

Ime: *	Prebivališče: *
Primék: *	Telefon: *
EMŠO: *	Mail: *
Datum rojstva: *	

Slika 30: Dodajanje zaposlenih.

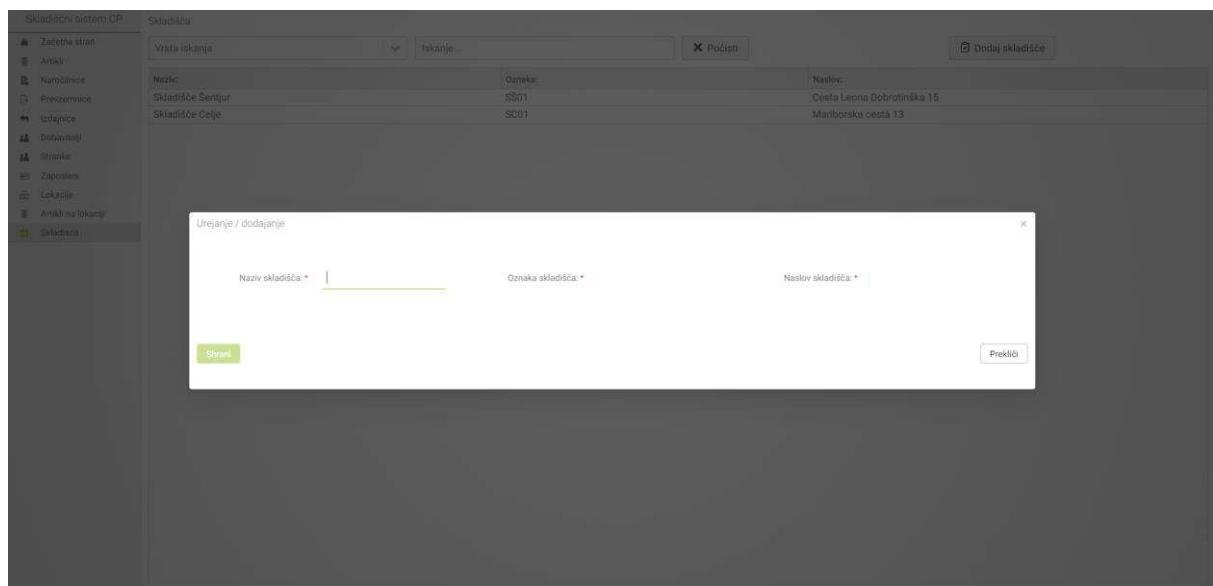
S klikom na osebo, katere podatke želimo spremeniti, se nam odpre okno, v katerem lahko posodobimo podatke in jih nato spet shranimo za nadaljnjo uporabo.



Slika 31: Urejanje zaposlenih.

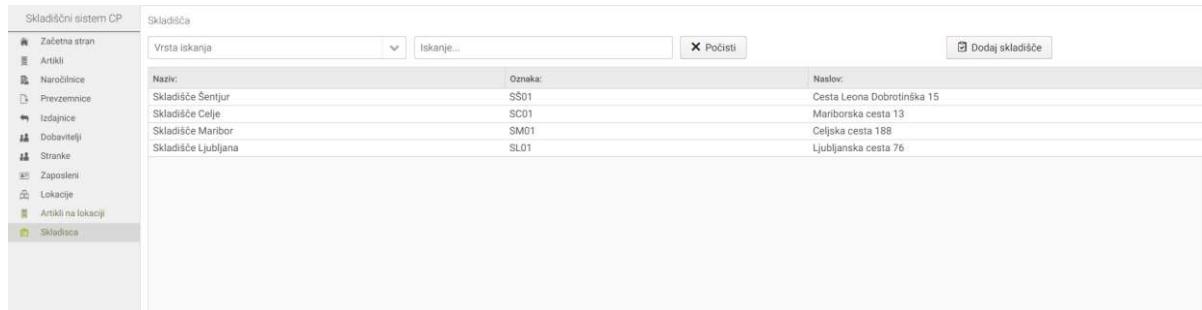
5.10 Skladišča

Pod zavihkom »skladišče« dodajamo skladišča. Tu skladišče opredelimo le z imenom in naslovom. Vsi ostali podatki se bodo dodajali in posodabljali pod zavihkom »lokacije v skladišču«.



Slika 32: Dodajanje skladišč.

Potrebni podatki so naziv, oznaka in naslov skladišča. V primeru, da je naziv skladišča Skladišče Šentjur, bo oznaka tega skladišča SŠ01, kar pomeni, da je to prvo skladišče v Šentjurju. To nam zagotavlja možnost kasnejšega širjenja skladišč v izbranem kraju.



The screenshot shows a search interface for warehouses. On the left is a sidebar with navigation links: Začetna stran, Artiki, Naročnice, Prevzemnice, Izdajnica, Dobavitelji, Stranke, Zaposleni, Lokacije, Artiki na lokaciji, and Skladišča. The main area has a header 'Skladišča' with dropdowns for 'Vrsta iskanja' (set to 'Skladišča') and 'Iskanje...', and a 'Počisti' button. A 'Dodaj skladišče' button is also present. Below is a table with columns: Naziv, Oznaka, and Naslov. The data is as follows:

Naziv	Oznaka	Naslov
Skladišče Šentjur	SŠ01	Cesta Leona Dobrotinške 15
Skladišče Celje	SC01	Mariborska cesta 13
Skladišče Maribor	SM01	Celjska cesta 188
Skladišče Ljubljana	SL01	Ljubljanska cesta 76

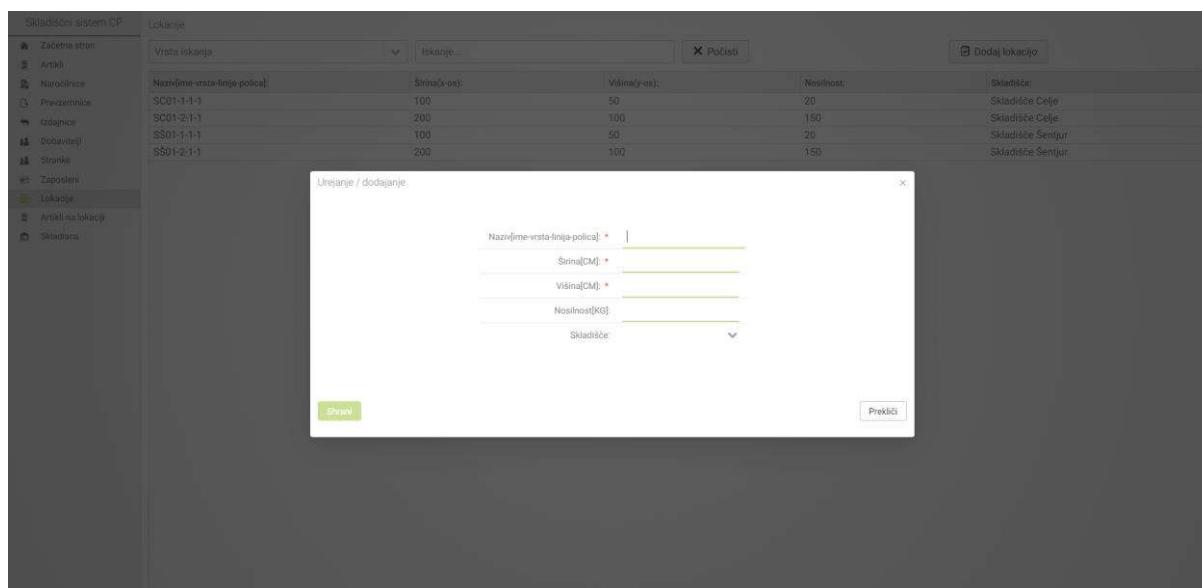
Slika 33: Seznam skladišč.

5.11 Lokacije v skladiščih

Lokacije v skladiščih dodatno razdelijo vsako skladišče na manjše dele. Za dodeljevanje prostorov potrebujemo naziv skladišča, enak kot tisti, ki smo ga določili pri ustvarjanju skladišč, na primer SŠ01. K temu imenu dodamo še oznake polic ali oddelkov. To lahko zapišemo kot SŠ01-x-y-z.

»x« predstavlja skladiščni prostor po širini, »y« prostor po dolžini in »z« prostor po višini.

Za konec dodamo še mere polic ali prostorov za skladiščenje, na seznamu poiščemo ime skladišča ter ga določimo.



The screenshot shows a search interface for locations. The sidebar is identical to the one in Screenshot 33. The main area has a header 'Lokacije' with dropdowns for 'Vrsta iskanja' (set to 'Lokacije') and 'Iskanje...', and a 'Počisti' button. A 'Dodaj lokacijo' button is also present. Below is a table with columns: Naziv/ime-vrsta/rinka/polic;. The data is as follows:

Naziv/ime-vrsta/rinka/polic;	Širina[cm];	Višina[cm];	Nosilnost;	Skladišče;
SC01-1-1-1	100	50	20	Skladišče Celje
SC01-2-1-1	200	100	150	Skladišče Celje
SŠ01-1-1-1	100	50	20	Skladišče Šentjur
SŠ01-2-1-1	200	100	150	Skladišče Šentjur

A modal window titled 'Urejanje / dodajanje:' is open, showing fields for 'Naziv/ime-vrsta/rinka/polic; *' (with 'SŠ01-1-1-1' entered), 'Širina[cm] *' (100), 'Višina[cm] *' (50), 'Nosilnost[KG]' (20), and a dropdown for 'Skladišče' (set to 'Skladišče Šentjur'). At the bottom are 'Shrani' and 'Prekliči' buttons.

Slika 34: Dodajanje lokacij v skladišču.

Lokacije					
Vrsta iskanja	Iskanje...		X Počisti		Dodaj lokacijo
Naziv(me vrsta linija polica):	Širina(x-os):	Vihina(y-os):	Nosilnost:	Skladišče:	
SC01-1-1-1	100	50	20	Skladišče Celje	
SC01-2-1-1	200	100	150	Skladišče Celje	
SS01-1-1-1	100	50	20	Skladišče Šentjur	
SS01-2-1-1	200	100	150	Skladišče Šentjur	
SC01-3-1-1	500	500	5000	Skladišče Celje	
SC01-4-1-1	200	250	1000	Skladišče Celje	
SS01-3-1-1	250	250	500	Skladišče Šentjur	
SS01-4-1-1	500	500	5000	Skladišče Šentjur	
SM01-1-1-1	100	100	250	Skladišče Maribor	
SM01-2-1-1	200	200	250	Skladišče Maribor	
SM01-3-1-1	500	500	1500	Skladišče Maribor	
SM01-4-1-1	500	500	7500	Skladišče Maribor	
SL01-1-1-1	100	100	250	Skladišče Ljubljana	
SL01-2-1-1	200	200	250	Skladišče Ljubljana	
SL01-3-1-1	500	500	5000	Skladišče Ljubljana	
SL01-4-1-1	500	500	7500	Skladišče Ljubljana	

Slika 35: Primer označevanja lokacij.

5.12 Artikli v skladiščih

Kot zadnje dejanje nam ostane dodajanje artiklov na vse prej ustvarjene lokacije po skladiščih. Na spustnem seznamu poiščemo izdelek in ga izberemo, določimo tudi količino, ki jo bomo shranili na to lokacijo, dodamo EM ter poiščemo in izberemo lokacijo skladišča.

Artikel:	Količina:	EM:	Lokacija:	Skladišče:
Cev B	5	KOS	SC01-2-1-1	Skladišče Celje
Cev B	5	KOS	SS01-2-1-1	Skladišče Šentjur
Vijak	500	KOS	SC01-1-1-1	Skladišče Celje
Vijak	500	KOS	SS01-1-1-1	Skladišče Šentjur

Urejanje / dodajanje

Artikel: *

Količina: *

EM: *

Lokacija: *

Izbrani

Preklici

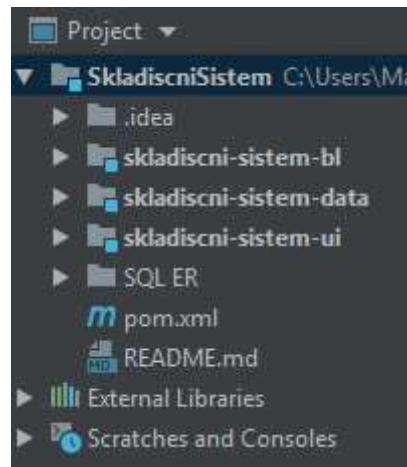
Slika 36: Dodajanje artiklov na lokacije.

Skladiščni sistem CP					
Artikli na lokacijah					
	nazivArtikla	Iskanje...	X Počisti	Dodaj artikel na lokacijo	
Artikel:	Količina:	EM:	Lokacija:	Skladišče:	
Cev B	5	KOS	SC01-2-1-1	Skladišče Celje	
Cev B	5	KOS	SS01-2-1-1	Skladišče Šentjur	
Vijak	500	KOS	SC01-1-1-1	Skladišče Celje	
Vijak	500	KOS	SS01-1-1-1	Skladišče Šentjur	
PVC kanal 10x10	250	KOS	SM01-1-1-1	Skladišče Maribor	
PVC kanal 10x10	250	KOS	SL01-2-1-1	Skladišče Ljubljana	
PVC kanal 10x10	500	KOS	SC01-2-1-1	Skladišče Celje	
Jupol classic RUMENA	50	KOS	SS01-3-1-1	Skladišče Šentjur	
Ploščati čopič 3/1	20	KOS	SS01-4-1-1	Skladišče Šentjur	
Kabelske vezice	150	KOS	SS01-2-1-1	Skladišče Šentjur	
Guma 295/80R 22.5	50	KOS	SS01-3-1-1	Skladišče Šentjur	
Rum domaći 40 vol%	100	KOS	SS01-3-1-1	Skladišče Šentjur	
Viljamovka 30%	50	KOS	SM01-1-1-1	Skladišče Maribor	
Viski Jameson 40 vol%	100	KOS	SL01-3-1-1	Skladišče Ljubljana	
Ploščati čopič	50	ZABOJ	SC01-4-1-1	Skladišče Celje	
Guma 295/80R 22.5	25	KOS	SL01-4-1-1	Skladišče Ljubljana	
Čokoladni liker 18 vol%	10	KOS	SM01-1-1-1	Skladišče Maribor	
Guma 295/80R 22.5	20	KOS	SM01-3-1-1	Skladišče Maribor	
Lesetenčna sponka	250	KOS	SC01-2-1-1	Skladišče Celje	
Priklična spojka 2273	60	KOS	SC01-2-1-1	Skladišče Celje	
Povezovalna sponka	100	KOS	SM01-4-1-1	Skladišče Maribor	
PVC cev 16"	15	KOS	SS01-2-1-1	Skladišče Šentjur	
Elektro vezice 50x	5	ZABOJ	SS01-4-1-1	Skladišče Šentjur	
Prtipoplavne vrteče 50kg	3	PALETA	SC01-4-1-1	Skladišče Celje	
Prtipoplavne vrteče 20kg	3	PALETA	SL01-3-1-1	Skladišče Ljubljana	
Vtična sponka	25	KOS	SM01-4-1-1	Skladišče Maribor	

Slika 37: Seznam artiklov v skladiščih.

6 KODA PROGRAMA

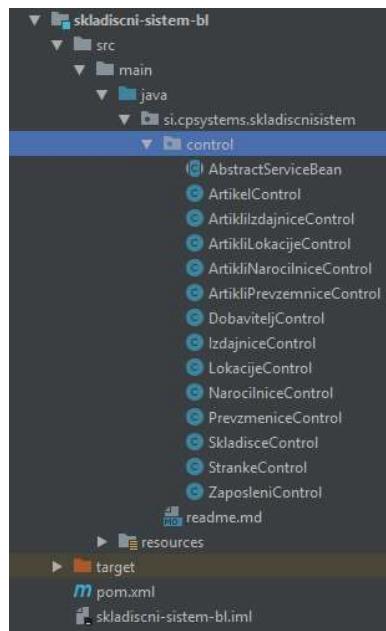
Najin projekt sva razdelila na tri module BL, data in UI. V BL so napisane metode za manipuliranje z entitetami. Data vsebuje enume, konverterje, adapterje, interface, entitete in pojo datoteke (priprava za PDF poročila). UI pa vsebuje vse zavihke, forme in ostale razrede, ki so uporabljeni v projektu.



Slika 38: Moduli projekta.

6.1 BL modul

Tukaj so napisane vse metode, ki se pokličejo preko vmesnikov na data modulu. Vsi razredi vsebujejo en abstraktni razred, kjer so napisane osnovne metode, kot so vstavljanje v bazo, brisanje, spremjanje, iskanje in ostalo.



Slika 39: Razredi v modulu BL.

Vse entitete uporablja tudi attribute »version« in »skrit«, kar pomeni da se ob vsaki spremembi ali ustvarjanju vrstice version poveča oziroma spremeni, ter pa skrbi za več uporabniško spremjanje vrstic, to pomeni, da se vedno vpiše najnovejši zapis. Atribut skrit pa skrbi za skrivanje vrstic pri prikazovanju in je tipa bool.

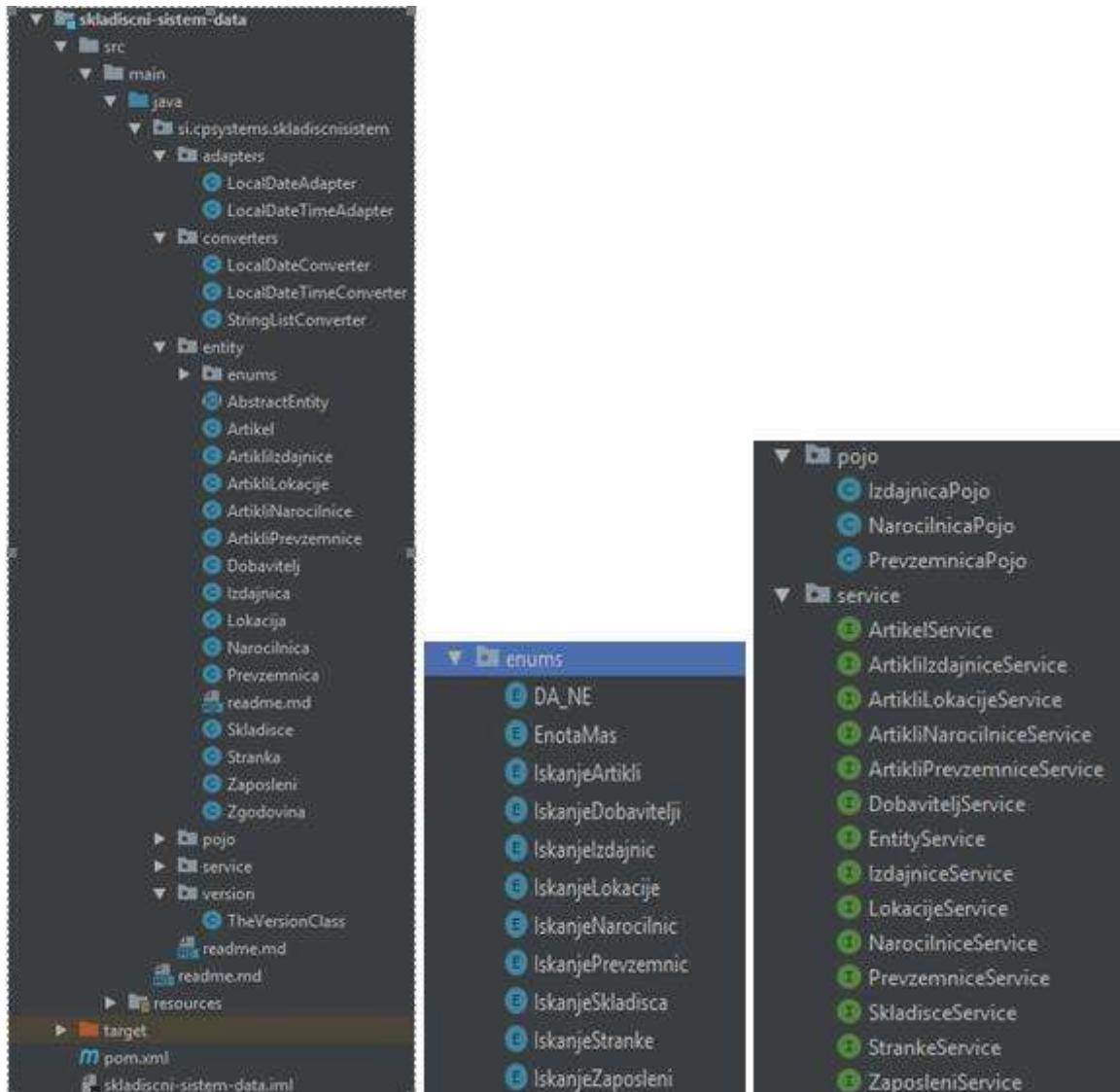
Vsek kontrol razred im za vsako entiteto svoje specialne metode, ki jih uporablja za računanje, prikazovanje, iskanje, spremjanje in podobno.

```
/*
 * Metoda za iskanje po nazivu
 *
 * @param naziv
 * @return vrne vse Article s danim nazivom
 */
@Override
public Article getByNaziv(String naziv) {
    List<Article> resultList = entityManager.createNamedQuery(Artikel.FIND_ALL, Artikel.class).getResultList();
    resultList = resultList.stream()
        .filter(Artikel -> naziv.equals(Artikel.getNazivArtikla()))
        .collect(Collectors.toList());
    if (resultList.size() == 1)
        return resultList.get(0);
    else
        return null;
}
```

Slika 40: Primer metode v razredu control.

6.2 DATA modul

Data modul vsebuje adapterje za vpisovanje podatkov v bazo, kot so datumi in datumi z uro, konverterje za razne tipe, ki jih vpisujemo v bazo, potem entitete in razred za abstraktne podatke v entiteti, enume, ki se uporablja za spustne sezname, pojoti za ustvarjanje JSON datotek, iz katerih se ustvari PDF dokument in servisov (interfaci) za metode na BL modulu.



Slika 41: Razredi, enumi, pojoti in vmesniki na modulu data.

6.2.1 Razredi entitet

Vsaka entiteta vsebuje svoje atribute in relacije, ki uporabljajo razne adapterje in converterje, svoje konstruktorje (prevzeti in pretvorbeni) in lastnosti (get in set), vsaka entiteta pa že ima vnaprej napisane sql ukaze za hitrejše delovanje.

```
@NotNull(message = "polje kolicina ne sme biti manjše od 0!")
@Column(name = "kolicina", nullable = false)
private Long kolicina;
```

Slika 42: Ustvarjanje atributa v entiteti.

```
public Artikel() { }

public Artikel(String sifraArtikla, String nazivArtikla, String znamkaArtikla, EnotaMas EM, Long kolicina,
    LocalDate datumVeljavnosti, BigDecimal teza, BigDecimal dolzina, BigDecimal sirina, BigDecimal visina,
    List<ArtikliNarocilnice> artikliNarocilniceList, List<ArtikliPrevzemnice> artikliPrevzemniceList,
    List<ArtikliLokacije> artikliLokacijeList, List<ArtikliIzdajnice> artikliIzdajniceList) {
    this.sifraArtikla = sifraArtikla;
    this.nazivArtikla = nazivArtikla;
    this.znamkaArtikla = znamkaArtikla;
    this.EM = EM;
    this.kolicina = kolicina;
    this.datumVeljavnosti = datumVeljavnosti;
    this.teza = teza;
    this.dolzina = dolzina;
    this.sirina = sirina;
    this.visina = visina;
    this.artikliNarocilniceList = artikliNarocilniceList;
    this.artikliPrevzemniceList = artikliPrevzemniceList;
    this.artikliLokacijeList = artikliLokacijeList;
    this.artikliIzdajniceList = artikliIzdajniceList;
}
```

Slika 43: Konstruktor entitete.

```
public Long getKolicina() { return kolicina; }

public void setKolicina(Long kolicina) { this.kolicina = kolicina; }
```

Slika 44: Lastnosti atributa.

6.2.2 Pojoti

Vsi pojoti, ki skrbijo za polnjenje JSON datotek delujejo na enak način, in sicer so to samostojni razredi, ki pridobijo podatke iz baze in jih zapisejo v JSON obliki. Ker pa je zaradi relacij potrebno poskrbeti, da nimamo ponavljačih se podatkov v JSON datoteki, je potrebno počistiti nekatere podatke, da dobimo le tiste, ki so ključni za uporabo.

```
public NarocilnicaPojo(Narocilnica narocilnica) {
    narocilnica.getZaposleni().getIzdajnicaList().clear();
    narocilnica.getZaposleni().getNarocilnicaList().clear();
    narocilnica.getZaposleni().getPrevzemnicaList().clear();
    narocilnica.getDobavitej().getNarocilnicaList().clear();
    narocilnica.getDobavitej().getPrevzemnicaList().clear();
    narocilnica.getPrevzemnicaList().clear();
    this.narocilnica = narocilnica;
    this.zaposleni = narocilnica.getZaposleni();
    this.dobavitej = narocilnica.getDobavitej();
}
```

Slika 45: Konstruktor pojota naročilnice.

6.2.3 Vmesniki (servisi)

Vmesniki, ki skrbijo za dostopanje do metod, so napisani na enak način, se pravi ustvarimo neko metodo, ji dodamo tip in vhodne podatke, napišemo pa jo na BL modulu.

```
@Local
public interface ArtikelService extends EntityService<Artikel> {

    Artikel getBySifra(Long sifra);

    Artikel getByNaziv(String naziv);

    Artikel getByKolicina(int kolicina);

    Long getLastSifra();

    List<Artikel> getKolicinaMIN(String vrsta, String vrednost);
}
```

Slika 46: Vmesnik (servis).

6.2.4 Enumi

Enumi so uporabljeni v spustnih menijih, kjer imamo neko omejeno izbiro oziroma določene vrednosti, ki jih lahko vstavimo.

```
public enum EnotaMas {
    KOS( value: "KOS"),
    ZABOJ( value: "ZABOJ"),
    PALETA( value: "PALETA");

    private String value;

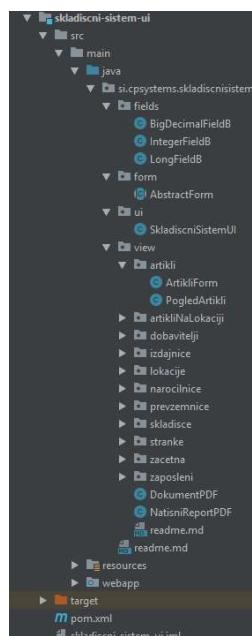
    EnotaMas( String value) {this.value = value; }

    public String value() {return value;}
}
```

Slika 47: Enum za enoto mase.

6.3 UI modul

UI modul ali uporabniški vmesnik vsebuje razrede za posebna tekstovna polja na formah, abstraktni razred za vse forme, glavni razred SkladiscniSistemUI, ki skrbi za sejo, piškotke, spremembo porta, spremjanje URL-ja ter meni. UI modul vsebuje tudi mapo za vsak zavihek, vsaka mapa pa vsebuje razred s pogled zavihka in formo na zavihku. Najbolj posebna razreda sta DokumentPDF in NatisniReportPDF, ki sta razreda za ustvarjanje JSON datotek, odpiranje PDF predogleda, povezovanja na JASPER server in ustvarjanje PDF dokumentov.



Slika 48: Razredi na modulu UI.

6.3.1 Primarni razred za UI

To je razred, v katerem so metode, ki skrbijo za URL aplikacije, port, varnost, sejo, piškotke, meni in ostale stvari, ki so potrebne v brskalnikih.

```
// Preusmeritev http na https
if (Page.getCurrent().getLocation().getScheme().equals("http")) {
    getUI().getPage().setLocation(Page.getCurrent().getLocation().toString().replace("target: \"http://\"", "replacement: \"https://\"").replace("target: \"8080\"", "replacement: \"8181\""));
}
Locale siLoc = new Locale(language: "SI", country: "SI");
setLocale(siLoc);
VaadinSession.getCurrent().setLocale(siLoc);
Locale.setDefault(siLoc);

VaadinSession.getCurrent().getSession().setMaxInactiveInterval(119 * 60);
Responsive.makeResponsive(...components: this);
addStyleName(ValoTheme.UI_WITH_MENU);
UI.getCurrent().getReconnectDialogConfiguration().setDialogText("Povezava je izgubljena, povezujem...");
getContent().setVisible(true);
```

Slika 49: Metoda za spreminjanje porta in protokola ter sledenje seje.

```
@.WebServlet(value = "/*", asyncSupported = true)
@VaadinServletConfiguration(productionMode = true, ui = SkladischniSistemUI.class, heartbeatInterval = 240, closeIdleSessions = true)
public static class Servlet extends VaadinCDIServlet {
    @Override
    protected final void servletInitialized() throws ServletException {
        super.servletInitialized();

        getService().setSystemMessagesProvider(
            systemMessagesInfo -> {
                CustomizedSystemMessages messages =
                    new CustomizedSystemMessages();
                messages.setSessionExpiredCaption("Seja je potekla");
                messages.setSessionExpiredMessage("Za nadaljevanje kliknite tukaj ali pritisnite tipko ESC.");
                return messages;
            });
    }
}
```

Slika 50: Metoda za obnovitev seje.

6.3.2 Pogledi zavihkov

Vsi pogledi zavihkov so razdeljeni na tri dele: inicializacijo, vstopno metodo in metodo ob prvem vstopu. Pod inicializacijo spadajo vse nastavitev vseh komponent, ki jih bomo uporabljali pri našem pogledu (servisi za dostopanje do metod, gumbi, tabele, tekstovna polja).

```
@CDIView(VIEW_NAME)
@ViewMenuItem(title = "Artikli", icon = VaadinIcons.ARCHIVES, order = 2)
@UIScoped
public class PogledArtikli extends FPanel implements View {
    public static final String VIEW_NAME = "artikli";
    @Inject
    private ArtikelService artikelService;
    @Inject
    private ArtikliForm artikliForm;
    @Inject
    private ArtikliLokacijeService artikliLokacijeService;

    private FVerticalLayout vsebina = new FVerticalLayout().withFullSize();
    private FLabel naslov = new FLabel("Artikli").withFullWidth().withStyleName("naslov");
    private FHorizontalLayout h1 = new FHorizontalLayout().withFullWidth();
    private FComboBox<IskanjeArtikli> iskanjeSeznam = new FComboBox<>().caption(null,
        Arrays.asList(IskanjeArtikli.values())).withFullSize().withPlaceholder("Vrsta iskanja");
    private FTextField iskanjeVrstica = new FTextField().withFullSize().withPlaceholder("Iskanje...");
    private FHorizontalLayout h2 = new FHorizontalLayout().withFullSize();
    private FGrid<Artikel> artikelFGrid = new FGrid<>(Artikel.class).withFullSize();
    private DateTimeFormatter formatter = DateTimeFormatter.ofPattern("dd.MM.yyyy");
    private FButton pocisti = new FButton("Pocisti").withIcon(VaadinIcons.CLOSE).withClickListener(event -> {
        iskanjeVrstica.setValue("");
        iskanjeSeznam.setValue(IskanjeArtikli.NASIV);
        artikelFGrid.setItems(artikelService.getAll());
    });
    private FButton dodajArtikel = new FButton("Dodaj artikel").withIcon(VaadinIcons.CLIPBOARD_CHECK);
}
```

Slika 51: Inicializacija komponent.

Vstopna metoda se proži ob vsakem vstopu na zavihek: sem spadajo stvari, kot so osveževanje tabel in podobno.

```
@Override
public void enter(ViewChangeListener.ViewChangeEvent event) {
    artikelFGrid.setItems(artikelService.getAll());

    dodajArtikel.addClickListener(event1 -> {
        artikliForm.setEntity(new Artikel());
        artikliForm.openInModalPopup();
    });
}
```

Slika 52: Metoda za vsak vstop.

Metoda ob prvem vstopu se sproži le, ko se aplikacija prvič požene in je potrebno naložiti vse komponente na zavihek (pogled). Tukaj se tudi zaženejo razne metode za delovanje forme, poslušalci za klike gumbov in spreminjanje vrednosti v tekstovnih poljih.

```
@PostConstruct
private void init() {

    artikliForm.setSavedHandler(this::save);
    artikliForm.setResetHandler(this::reset);

    artikelFGrid.addItemClickListener((ItemClickListener<Artikel>) eventl -> {
        artikliForm.setEntity(eventl.getItem());
        artikliForm.openInModalPopup();
    });

    iskanjeSeznam.addValueChangeListener(event -> {
        if (event != null) {
            if (iskanjeSeznam.getValue() == null) {
                iskanjeSeznam.setValue(IskanjeArtikli.NAZIV);
            }
        }
    });
}

iskanjeVrstica.setValueChangeMode(ValueChangeMode.LAZY);
iskanjeVrstica.addValueChangeListener(e -> artikelFGrid.setItems(artikelService
    .getAll(iskanjeSeznam.getValue().toString(), iskanjeVrstica.getValue())));
iskanjeSeznam.addValueChangeListener(e -> artikelFGrid.setItems(artikelService
    .getAll(iskanjeSeznam.getValue().toString(), iskanjeVrstica.getValue())));

artikelFGrid.removeAllColumns();
artikelFGrid.addColumn(Artikel::getSifraArtikla).setCaption("Interni šifra");
artikelFGrid.addColumn(Artikel::getNazivArtikla).setCaption("Naziv");
artikelFGrid.addColumn(Artikel::getKolicina).setCaption("Količina");
artikelFGrid.addColumn(Artikel::getEM).setCaption("EM");
artikelFGrid.addColumn(Artikel::getDolzina).setCaption("Dolžina[MM]");
artikelFGrid.addColumn(Artikel::getSirina).setCaption("Širina[MM]");
artikelFGrid.addColumn(Artikel::getVisina).setCaption("Višina[MM]");
artikelFGrid.addColumn(Artikel::getTeza).setCaption("Teža[KG]");
artikelFGrid.addColumn(Artikel::getDatumVeljavnosti).setCaption("Datum veljavnosti")
    .setRenderer(new LocalDateRenderer(formatter));

artikelFGrid.withItems(artikelService.getAll());

h1.addComponent(iskanjeSeznam, iskanjeVrstica, pocisti, dodajArtikel);
h2.addComponent(artikelFGrid);
vsebina.withComponents(naslov, h1, h2).withExpandRatio(h2, ratio: 1);
withContent(vsebina).withFullSize();
}
```

Slika 53: Metoda za ob prvem vstopu.

6.3.3 Forme na zavihkih (pogledih)

Vsek pogled ima svojo formo za vstavljanje in spremanje atributov. Forma se odpre če to sproži gumb »ustvari novo« ali pa s klikom na nek podatek v tabeli. Če ustvarimo nov podatek, se nam odpre prazna forma, če izberemo nek atribut v tabeli, pa je forma že izpolnjena in pripravljena na spremembe.

Enako kot pri pogledih imamo tudi tukaj tri osnovne metode in dve dodatni. Prve tri so vstopna, inicialna in metoda ob prvem vstopu. Dodatni sta za ustvarjanje dodatnega okna na katerem se pokaže forma, in pa metoda za zapolnjevanje forme z že shranjenimi podatki.

```
public ArtikliForm() {  
    super(Artikel.class);  
    setWidth("80%");  
    setHeight("55%");  
}
```

Slika 54: Metoda za ustvarjanje forme.

```
@Override  
public void setEntity(Artikel artikel) { super.setEntity(artikel); }
```

Slika 55: Metoda za nastavljanje podatkov na formo.

6.3.4 Dodatni specialni razredi za delovanje UI modula

To je metoda za odpiranje predogleda PDF dokumenta, ki ga lahko kasneje natisnemo, shranimo, pošljemo in podobno.

```
public class DokumentPDF {  
    public static void openInModalPopup(byte[] bytes, String caption, String filename) {  
        boolean pdf = bytes != null && bytes.length >= 4; // && bytes[0] == 0x25 && bytes[1] == 0x50 && bytes[2] == 0x44 && bytes[3] == 0x46 && bytes[4] == 0x2D  
        if (pdf) {  
            UI.getCurrent().addWindow(  
                new MWindow(caption, new BrowserFrame( caption: null, new StreamResource(() ->  
                    new ByteArrayInputStream(bytes), filename: filename + "_" + UUID.randomUUID().toString()) ) {  
                    setSizeFull();  
                })  
                .withModal(true).withCenter().withDraggable(false).withWidth( width: 70, Sizeable.Unit.PERCENTAGE).withFullHeight());  
        }  
    }  
}
```

Slika 56: Metoda za odpiranje PDF pregledovalnika.

Metoda za generiranje JSON datotek, povezavo na JASPER server, vstavljanje datotek na PDF in generiranje PDF datoteke.

```
private String jsonString = "";  
private ObjectMapper mapper = new ObjectMapper();  
  
public void GenerirajNarocilnicaPDF(NarocilnicaPojo narocilnicaPojo) {  
    mapper.enable(SerializationFeature.WRAP_ROOT_VALUE);  
    mapper.registerModule(new JavaTimeModule());  
    try {  
        jsonString = mapper.writeValueAsString(narocilnicaPojo);  
    } catch (JsonProcessingException e) {  
        e.printStackTrace();  
    }  
    InputStream stream = new ByteArrayInputStream(jsonString.getBytes(Charset.forName("UTF-8")));  
    Map<String, Object> parameters = new HashMap<>();  
    parameters.put(JsonQueryExecuterFactory.JSON_DATE_PATTERN, "dd.MM.yyyy");  
    parameters.put(JsonQueryExecuterFactory.JSON_NUMBER_PATTERN, "#,##0.##");  
    parameters.put(JsonQueryExecuterFactory.JSON_LOCALE, Locale.getDefault());  
    parameters.put(JRParameter.REPORT_LOCALE, Locale.getDefault());  
    parameters.put(JsonQueryExecuterFactory.JSON_INPUT_STREAM, stream);  
    String sourceFile = "C:/Users/Matjaz/Desktop/SkladisniSistem/SSJasper/";  
    ByteArrayOutputStream bos = new ByteArrayOutputStream();  
    try {  
        mapper.writerWithDefaultPrettyPrinter().writeValue(new File( pathname: "C:/Users/Matjaz/Desktop/Narocilnica.json"), narocilnicaPojo);  
        JasperPrint jasperPrint = JasperFillManager.fillReport( sourcefileName: sourceFile + "Narocilnica.jasper", parameters);  
        JasperExportManager.exportReportToPdfStream(jasperPrint, bos);  
    } catch (JRException | IOException e) {  
        e.printStackTrace();  
    }  
    DokumentPDF.openInModalPopup(bos.toByteArray(), caption: "PDF", filename: "Narocilnica_PDF");  
}
```

Slika 57: Metoda za ustvarjanje in polnjenje PDF dokumenta.

7 ANKETA

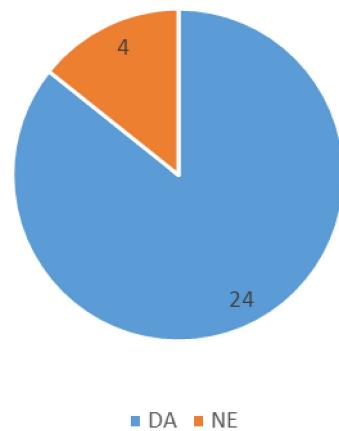
Izvedla sva anketo, v kateri sva anketirala dijake četrtega letnika Šolskega centra Celje. Izbrala sva naključne sošolce in jim predstavila anketo, še prej pa sva jim razložila osnovne stvari o skladiščnih sistemih, kaj so in kako delujejo. Ustrezno je anketo rešilo 28 dijakov in dijakinj. Z anketo sva pridobila odgovore na ključna vprašanja, ki sva jih potrebovala za nadaljevanje.



Graf 1: Ste že kdaj uporabljali skladiščni sistem?

Ugotovila sva, da večina anketirancev še ni uporabljala skladiščnega sistema, kar je bilo tudi pričakovano.

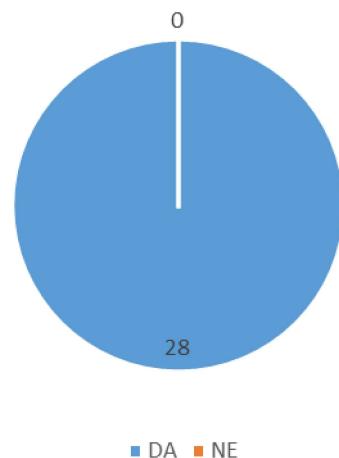
2. Ali veste kaj je skladiščni sistem?



Graf 2: Ali veste, kaj je skladiščni sistem?

Večina anketirancev ve, kaj je skladiščni sistem.

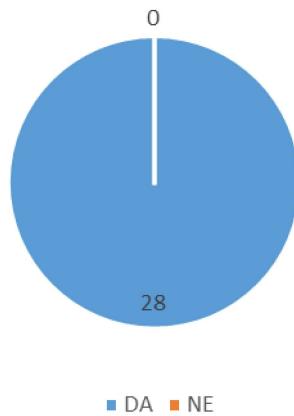
3. Ali se vam zdi skladiščni sistem uporaben?



Graf 3: Ali se vam zdi skladiščni sistem uporaben?

Odgovor je bil pričakovani, skladiščni sistem se vsem zdi uporaben.

4. Bi v svojem podjetju uporabili skladiščni sistem?



Graf 4: Bi v svojem podjetju uporabili skladiščni sistem?

Vsi anketiranci so enotno odgovorili na vprašanje, kar je bilo tudi pričakovati.

7.1 Zaključki ankete

Kljub majhnemu, nerepresentativnemu vzorcu, meniva, da so rezultati ankete dovolj informativni. Vsem anketircem so blizu računalniške teme, zato je verjetno mnogim od njih skladiščni sistem poznan.

Na najpomembnejši vprašanji, ali bi v svojem podjetju uporabili skladiščni sistem in ali se jim zdi skladiščni sistem uporaben, so vsi odgovorili z »da«, zato meniva, da je prihodnost takšnih spletnih aplikacij svetla in bi se lahko dobro tržila.

7.2 Ugotovitve glede na hipoteze

Med raziskovanjem in ustvarjanjem spletne aplikacije sva si zastavila pet hipotez, ki jih bova sedaj potrdila.

Hipoteza 1: večina anketirancev bo skladiščne sisteme označila kot učinkovite.

Hipoteza je v potrjenja, saj je le 6 anketirancev skladiščne sisteme označilo kot neučinkovite.

Hipoteza 2: za izdelavo skladiščnega sistema bo potrebno izvenšolsko znanje.

Hipoteza je potrejna, ker se vsega uporabljenega znanja nisva naučila v šoli.

Hipoteza 3: uporaba skladiščnega sistema bo učinkovita in hitra.

Hipoteza je potrjena, saj je spletna aplikacija hitra in odzivna.

Hipoteza 4: skladiščni sistem ustreza vsem vrstam skladišč.

Hipoteza je potrjena, saj je uporaba mogoča v vseh vrstah skladišč.

Hipoteza 5: aplikacijo bo mogoče nareediti brez stroškov.

Hipoteza je potrjena, ker za izdelavo spletne aplikacije nisva potrebovala nikakršnih dodatnih vložkov v strojno ali programsko opremo.

8 ZAKLJUČEK

Med raziskovalno nalogo sva prišla do veliko novih spoznanj in izkušenj na področju skladiščnih sistemov in programiranja. Potrebnega je bilo tudi veliko timskega dela, ki sva ga opravila brez težav.

Pri izdelavi spletne aplikacije sva naletela na težave, vendar sva jih z veliko truda in dela odpravila v najkrajšem možnem času. Dosegla sva cilj, ki sva si ga zastavila na začetku, ob izbiri teme skladiščnih sistemov, in ustvarila delajočo in uporabno spletno aplikacijo.

Znanje, ki sva ga pridobila ob izdelovanju, nama bo prišlo prav tudi pri nadaljnem šolanju in pri razvijanju in izdelavi novih spletnih aplikacij ter prodaji in razlagi pri morebitnih delodajalcih.

Najin program je pripravljen na nadaljno razvijanje in dodajanje funkcij. Ena od teh funkcij bi bila FIFO (first in first out), ki se uporablja za knjiženje. Aplikacijo sva zasnova na tak, da se jo lahko vključi v »namizje« vsakega podjetja, ki bi jo potrebovalo. To pomeni, da lahko najina spletna aplikacija deluje znotraj neke druge aplikacije kot samostojna, s svojimi povezavami do spletnih strežnikov in baz, ki jih zahteva za uporabo.

Pomembna razširitev, ki jo načrtujeva pri nadgradnji aplikacije je sistem oblaka. V oblak bi se dnevno shranjevale kopije podatkov iz glavnega strežnika in s tem ustvarjale arhiv podatkov, s čimer bi preprečila izgube.

Vsem podjetjem, ki se ukvarjajo s skladiščenjem, bi priporočala uporabo skladiščnega sistema, saj omogoča večjo preglednost in dodatne funkcionalnosti pri poslovanju.

9 VIRI IN LITERATURA

GitHub. (19. 2 2019). Pridobljeno iz Payara: <https://github.com/payara/Payara>

Matič, M. (16. 2 2019). Pridobljeno iz Digitalna knjižnica univerze v Mariboru:
<https://dk.um.si/Dokument.php?id=18197>

Minoa. (22. 2 2019). Pridobljeno iz <http://www.minoa.si/Default.aspx?ID=39>

Popović, S. (25. 2 2019). *BB.* Pridobljeno iz Skladiščno poslovanje v izbranem podjetju:
https://www.bb.si/doc/diplome/Popovic_Sandra.pdf

Rouse, M. (1. 3 2019). Pridobljeno iz MySQL:

<https://searchoracle.techtarget.com/definition/MySQL>

Wikipedija. (1. 3 2019). Pridobljeno iz Skladiščenje:

<https://sl.wikipedia.org/wiki/Skladi%C5%A1%C4%8Denje>

Wikipedija. (1. 3 2019). Pridobljeno iz POJO:

https://en.wikipedia.org/wiki/Plain_old_Java_object

Wikipedija. (27. 2 2019). Pridobljeno iz Visual Studio:

https://sl.wikipedia.org/wiki/Visual_Studio

Wikipedija. (1. 3 2019). Pridobljeno iz IntelliJ IDEA:

https://sl.wikipedia.org/wiki/IntelliJ_IDEA

Wikipedija. (1. 3 2019). Pridobljeno iz IntelliJ IDEA:

https://en.wikipedia.org/wiki/IntelliJ_IDEA

Wikipedija. (27. 2 2019). Pridobljeno iz Vaadin: <https://en.wikipedia.org/wiki/Vaadin>

Wikipedija. (20. 2 2019). Pridobljeno iz Payara Server:

https://en.wikipedia.org/wiki/Payara_Server

10 PRILOGE

10.1 Anketni vprašalnik

<p>www.1ka.si</p> <p>Pozdravljeni, sva dijaka 4. letnika in opravljava anketo o skladiščnih sistemih zato vas prosiva za sodelovanje. Hvala že v naprej!</p> <p>XSPOL - Spol:</p> <p><input type="radio"/> Moški <input type="radio"/> Ženski</p> <p>XSTAR2a4 - V katere starostno skupino spadate?</p> <p><input type="radio"/> do 20 let <input type="radio"/> 21 - 40 let <input type="radio"/> 41 - 60 let <input type="radio"/> 61 let ali več</p> <p>Q1 - Ste že kdaj uporabljali skladiščni sistem?</p> <p><input type="radio"/> DA <input type="radio"/> NE</p> <p>Q2 - Ali veste kaj je skladiščni sistem?</p> <p><input type="radio"/> DA <input type="radio"/> NE</p> <p>Q3 - Se vam zdi skladiščni sistem uporaben?</p> <p><input type="radio"/> DA <input type="radio"/> NE</p> <p>Q4 - Bi v svojem podjetju uvedli skladiščni sistem zaradi hitrejšega poslovanja?</p> <p><input type="radio"/> DA <input type="radio"/> NE</p>	<p><i>Skladiščni sistem</i></p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------

IZJAVA*

Mentor Boštjan Resinović v skladu z 2. in 17. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi z naslovom SKLADIŠČNI SISTEM, katere avtorja sta Matjaž Centrih in Žan Prahuta:

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljeni literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogu v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno naložno dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektrem citiraju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, 8. 3. 2019



Podpis mentorja

Podpis odgovorne osebe

POJASNILO

V skladu z 2. in 17. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.